

FACULTAD DE INFORMÁTICA DE BARCELONA



COMPARATIVA DE MÉTODOS DE LOCALIZACIÓN CON SMARTPHONES

TRABAJO FINAL DE GRADO

GRADO EN INGENIERÍA INFORMÁTICA

TECNOLOGÍAS DE LA INFORMACIÓN

PAU FERRER CID

DIRECTOR: DR. JORGE GARCÍA VIDAL

Resumen

Desde la aparición del iPhone en enero del 2007 y de los teléfonos Android en octubre del 2008, el uso de teléfonos inteligentes (*smartphones*) ha crecido de manera sostenida, de modo que ya forman parte de nuestro día a día.

Muchas de las aplicaciones que se desarrollan para smartphones requieren la localización de los usuarios (*Location-Based Applications*). Las librerías de soporte al desarrollo (*Software Development Kit*, SDK) de Android y de iOS proporcionan métodos de localización (por ejemplo: *LocationManager* en Android o *CLLocationManager* en iOS). Sin embargo, en muchas ocasiones las soluciones ofrecidas por estos SDKs no son óptimas, bien por una falta de precisión en la localización, o por un excesivo consumo de batería. Por ello, en este proyecto se investiga cómo se puede combinar la información proporcionada por los servicios estándares de los SDKs con la información de otros sensores (ej. acelerómetro, giroscopio), para así obtener técnicas de localización que pueden ser más útiles y eficientes en muchos casos y compararlas con las comúnmente usadas en la actualidad.

Resum

Des de l'aparició de l'iPhone el gener del 2007 i dels telèfons Android l'octubre del 2008, l'ús de telèfons intel·ligents (*smartphones*) ha crescut de manera sostinguda, de manera que ja formen part del nostre dia a dia.

Moltes aplicacions que es desenvolupen per a smartphones requereixen de la localització dels usuaris (*Location-Based Application*). Les llibreries de suport al desenvolupament (*Software Development Kit*, SDK) de Android i de iOS proporcionen mètodes de localització (per exemple: *LocationManager* en Android o *CLLocationManager* en iOS). Però, en moltes ocasions les solucions oferides per aquests SDKs no són òptimes, ja sigui per una falta de precisió en la localització, o per un excessiu consum de bateria. Per això, en aquest projecte s'investiga com es pot combinar la informació proporcionada pels serveis estàndards dels SDK amb la informació d'altres sensors (ex: acceleròmetre, giroscopi), per així obtenir tècniques de localització que poden ser més útils i eficients en molts casos i comparar-les amb les normalment utilitzades en l'actualitat.

Abstract

Since the release of the iPhone in January 2007 and the release of the Android phones in October 2008, the use of the smartphones has grown steadily, so that they are already part of our daily life.

Lots of applications developed for smartphones require the users localization (Location-Based Application). The Android and iOS software development kits (SDK) offer localization methods (for example: *LocationManager* for Android and *CLLocationManager* for iOS). But in most cases the solution offered by the SDKs is not optimal, either because they lack of precision or they have an excessive power consumption. For that, this project investigates how we can combine the information given by the standard services of the SDKs with other sensor information (for example: accelerometers and gyroscopes), in order to obtain localization techniques that can be more useful and more efficient in a lot of cases and compare them with the commonly used nowadays.

Agradecimientos

Primero dar gracias a los profesores Jorge García Vidal y Jose María Barceló por guiarme y ayudarme durante el proyecto y por haberme dado la oportunidad de colaborar con el grupo de investigación SANS. También dar las gracias a todos los miembros del grupo por ayudarme en todo lo posible. Sobre todo dar las gracias a mi familia, a mi madre, padre, hermano y abuelos, sin ellos nada de esto habría sido posible. Gracias por vuestro apoyo incondicional y por haberme dado ánimos en los días en los que lo veía todo oscuro.

Índice general

Índice de tablas

Índice de figuras

1. Introducción	2
1.1. Formulación del Problema	2
1.2. Contexto	4
1.3. Estado del Arte	5
1.4. Alcance	7
2. Gestión del Proyecto	8
2.1. Metodología y Rigor	8
2.2. Planificación Temporal	9
2.2.1. General	9
2.2.2. Tareas	9
2.2.3. Recursos	11
2.2.4. Desviaciones	11
2.3. Gestión económica	13
2.3.1. Estimación de Costes	13
2.3.2. Control de gestión	15
3. Sostenibilidad y compromiso social	16

3.1. Impacto ambiental	16
3.2. Impacto económico	17
3.3. Impacto social	17
4. Sistemas de Localización Outdoor	19
4.1. GPS	19
4.2. Redes Celulares	20
4.3. Sistemas de Coordenadas Geográficas	21
4.3.1. Sistema LLA	21
4.3.2. Sistema ECEF	21
4.3.3. Sistema NED	21
4.4. Localización en Android	22
5. IMU	24
5.1. Sensores	25
5.2. Sistema de Coordenadas Local/body	26
5.3. Sensores en Android	26
6. Fusión de Sensores	28
6.1. Filtro de Kalman	28
6.2. Filtro de Kalman Extendido	32
7. Implementación: Recolección de datos	33
7.1. Pre-procesado de datos	33
8. Implementación: Modelo INS	36
8.1. Ecuaciones de cinemática	36
8.2. Transformación sistema de coordenadas	38

9. Implementación: Filtro de Kalman	41
9.1. Estructura	41
9.2. GPS/INS	44
9.3. Red Celular/INS	48
10.Tests	49
10.1. GPS/INS	49
10.2. Red Celular/INS	54
10.3. Consumo	56
11.Conclusiones	58
12.Trabajo futuro	61
13.Valoración Personal	62
Glosario	63
A. Elipses de Error	65
B. Transformación de coordenadas inversa	66
Bibliografía	68

Índice de tablas

2.1. Tareas	10
2.2. Distribución horas Gantt	13
2.3. Costes humanos	13
2.4. Costes hardware	14
2.5. Costes totales	14
3.1. Matriz de sostenibilidad	18
4.1. Datos FusedLocationProvider (*Posibles Métodos)	23

Índice de figuras

1.1. Resultado artículo detección de accidentes	5
4.1. Sistemas de coordenadas ECEF y NED	22
5.1. Azimut	24
5.2. Sistema de Coordenadas dispositivo Android	26
6.1. Proceso de Markov	29
6.2. Paso predicción Filtro de Kalman	31
6.3. Paso corrección Filtro de Kalman	31
7.1. Filtrado módulo acelerómetro	34
7.2. Ejemplo datos acelerómetro	35
7.3. Ejemplo datos giroscopio	35
8.1. Ejes giroscopio	37
8.2. Ejemplo transformación Body-NED	38
9.1. Estructura integración filtro de Kalman	42
9.2. Diagram de flujo del Kalman	43
9.3. Ejemplo gráfico incertidumbre introducida por los sensores . . .	47

10.1. Ejemplo trayecto realizado	49
10.2. Estimación navegación con INS	50
10.3. Estimación navegación con fusión de sensores	51
10.4. Muestra giro trayecto	51
10.5. Error vs. Período muestreo GPS (GPS/INS)	52
10.6. Simulación error en GPS	53
10.7. Simulación pérdida señal GPS	54
10.8. Solución navegación Red Celular/INS	55
10.9. Error vs. Período muestreo Red Celular (Red celular/INS) . . .	56
10.10Consumo vs. Período muestreo GPS	57
11.1. Error vs. Consumo	60
1. Diagrama Gantt	64

1 | Introducción

1.1. Formulación del Problema

Una parte muy importante de las aplicaciones para teléfonos inteligentes utiliza la localización de usuarios mediante sus móviles. Algunos ejemplos de este tipo de aplicaciones son: las aplicaciones de navegación, juegos, etc.

Tanto el sistema Android como el iOS incorporan librerías que permiten localizar los dispositivos en exteriores de edificios (localización *outdoor*). Los métodos utilizados se basan en el uso del GPS (*Global Positioning System*), o en la localización por red celular o por puntos de acceso WiFi.

Obviamente, el hecho de que estas librerías estén incorporadas de forma nativa en el sistema operativo ofrece muchas ventajas para el programador. Sin embargo, tal como se verá en esta memoria, en muchos casos las funcionalidades ofrecidas por dichas librerías no son las que necesita la aplicación. Normalmente esto ocurre porque, o bien la precisión de los resultados no es la deseada, o porque el consumo energético es demasiado grande. Más en concreto, el GPS permite tener una precisión mucho mayor que el uso de la localización por la red celular, pero también requiere un consumo de energía mayor. Además de estos motivos técnicos, aparecen otro tipo de problemas, principalmente debidos a que las tiendas de aplicaciones móviles (*Google Play* o *Apple Store*) tienen políticas restrictivas a la hora de publicar aplicaciones que usan la localización GPS.

El objetivo principal de este proyecto es estudiar, de forma experimental, hasta qué punto es viable incorporar las medidas de sensores inerciales normalmente presentes en los smartphones (por ejemplo, el acelerómetro y el giroscopio), de forma que aplicando una técnica de fusión de sensores, se puedan obtener métodos de localización mejor adaptados a las necesidades de las aplicaciones.

Más en concreto, se estudiará el uso del filtro de Kalman, que es una técnica

clásica para resolver el problema de la fusión de sensores.

El trabajo se centra en la localización de dispositivos Android montados en vehículos en movimiento, lo que acota el tipo de escenarios que debemos considerar.

De forma experimental, se comparan los diferentes compromisos que surgen a la hora de mejorar la precisión de la localización, sin incrementar el consumo energético.

Esta comparativa urge, ya que el consumo de las aplicaciones se ha convertido en un elemento crítico para el usuario, además del ahorro energético involucrado. Así, se pretende ver como los diferentes métodos afectan al consumo energético.

Entre los objetivos están:

1. Estudiar diferentes opciones de localización con móviles (GPS, Redes Celulares, Sensores inerciales y filtro de Kalman) .
2. Implementar dichas estrategias.
3. Hacer una comparación de los métodos utilizados con datos reales para poder comparar:
 - Su precisión.
 - Su consumo energético.

1.2. Contexto

Desde el año 2007 (fecha de la presentación del iPhone) a la actualidad, la utilización de aplicaciones móviles se ha generalizado en casi todos los segmentos sociales. En numerosas aplicaciones móviles la localización juega un papel fundamental. Por ejemplo, el uso de la localización permite desarrollar nuevos tipos de juegos, mientras que también permite desarrollar aplicaciones más críticas como la detección de accidentes de coches.

Las técnicas de navegación inercial - basadas en la información proporcionada por sensores tales como acelerómetro, magnetómetro y giroscopio - han sido objeto de estudio durante años, dado su bajo consumo energético respecto al GPS. En el caso que nos ocupa, es de interés estudiar cómo se puede fusionar la información obtenida a partir de dichos sistemas de navegación inercial con la información de posicionamiento proporcionada por el GPS o la localización por redes celulares.

Este problema, la fusión de sensores, es un problema clásico en los sistemas de navegación. De hecho, la primera aplicación del filtro de Kalman - creado por Rudolf E. Kalman y Richard S. Bucy en los años 60 - se hizo en el programa aeroespacial norteamericano Apolo para fusionar sensores presentes en la nave espacial, y así aterrizar correctamente.

Los principales beneficiarios de este proyecto son los desarrolladores de aplicaciones móviles que utilizan geolocalización. El proyecto estudia las diferentes opciones que tienen para incorporar un método de localización u otro según las necesidades de la aplicación, e incluso utilizar el diseño de los diferentes métodos realizados en este proyecto.

Este proyecto se desarrolla en el grupo de investigación SANS de la Facultad de Informática de Barcelona. A parte de hacer la implementación y comparación de métodos, el proyecto pretende ser un puente entre el estudio de la movilidad y el proyecto Captor (sensores de contaminación) del grupo.

1.3. Estado del Arte

En la literatura se han publicado numerosos estudios que tratan sobre los métodos de navegación inercial y su fusión con otros métodos de geolocalización, tales como el GPS.

Entre los diferentes trabajos hechos hasta el momento encontramos una tesis sobre la localización de personas dentro de edificios [21]. En esta tesis se puede observar como se hace uso de un sistema de navegación inercial (con acelerómetro y giroscopio) fusionado con otros sensores para mantener la localización. En nuestro caso no estudiamos el uso de sensores inerciales para la localización dentro de edificios (*indoor*), pero las ideas presentadas en dicha tesis son de interés para este trabajo.

En un artículo [1], los autores utilizan un pequeño dispositivo dotado de acelerómetro y GPS para poder detectar la posición de un vehículo en caso de que se produzca un accidente. Utilizan un filtro de Kalman, fusionando así los datos del acelerómetro con los del GPS. Existen dos razones por las que utilizan dicha estrategia, porque consiguen una localización precisa y sobre todo porque les permite conocer de modo aproximado la situación del dispositivo, pese no haber recibido señal del GPS durante unos cuantos segundos. Este artículo es de especial relevancia para nuestro trabajo, aunque no se sabe si los autores utilizan el API de Google para usar la localización mediante GPS.

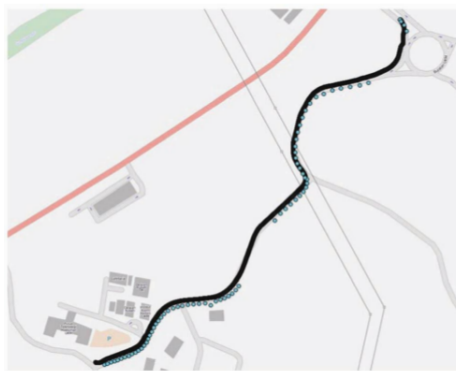


Figura 1.1: Imagen extraída de [1]. Se muestra en azul la posición GPS y en negro la posición derivada con el filtro.

Otra investigación [3] muestra otro problema de fusión de sistema de navegación inercial (INS) y GPS. Utiliza un filtro de Kalman, pero con una estructura *Indirect Feedback*, que estima el error en la posición y velocidad para corregir el INS. Entre sus resultados se puede ver como realmente la fusión de datos consigue reducir el error del INS.

En [16] se comparan las arquitecturas *Feedback* y *Feedforward*. En la segunda se estiman las posiciones y velocidades, en cambio en la primera se estiman los errores para poder corregir el INS.

Pese a que la opción más utilizada para la fusión de sensores es el filtro Kalman, también existen trabajos [10] en los que se utiliza un filtro de Partícula¹ para realizar la fusión. En nuestro caso, esta opción queda descartada dado que se necesitaría un número alto de partículas y el coste computacional sería demasiado alto para un dispositivo móvil.

Existe otro método de fusión de sensores que no utiliza medidas externas del GPS o red celular. El método Madgwick [12] realiza una fusión del acelerómetro y giroscopio con el magnetómetro de forma eficiente. Además, este método puede utilizarse con períodos de muestreo bajos. Este método es muy conocido en el mundo de los drones, donde la mayoría de aplicaciones de navegación utilizan esta técnica.

En la mayoría de artículos se utilizan sistemas dotados de un microcontrolador y sensores para hacer la fusión de sensores y así poder obtener un mejor sistema de localización. Por eso, este trabajo difiere de la mayoría en el hecho que no se utiliza ningún dispositivo construido específicamente para el estudio, sino que se utilizan los propios sensores de un smartphone. Una excepción es un trabajo [8], que utiliza la API de Google para obtener la localización del GPS (la misma que se utiliza en este proyecto). En dicho artículo se muestra un buen resultado de fusión mediante el filtro de Kalman pero utiliza un período de muestreo² muy pequeño, lo cual puede afectar gravemente al consumo de batería.

No se han encontrado trabajos en los que se estudie la fusión de los sensores inerciales con las localización basada en redes celulares.

¹Filtro de Partícula: método de estimación Bayesiano que utiliza métodos secuenciales de Monte-Carlo.

²Período de muestreo: tiempo entre recogida de muestras de un sensor.

1.4. Alcance

El alcance de esta comparativa es poder dar unos datos de referencia a desarrolladores de aplicaciones móviles que usan la geolocalización. Mostrar las diferentes opciones que tienen, con sus ventajas e inconvenientes. Cada una de estas opciones (navegación inercial o fusión de datos por ejemplo) podría ser implementada por ellos utilizando los mismos algoritmos usados en este proyecto u otros.

Otro objetivo del alcance es llegar al máximo numero de desarrolladores que trabajan con aplicaciones móviles y hacen uso de la geolocalización. Para que así, puedan tener presente otras técnicas que no vienen por defecto en los lenguajes de programación para teléfonos móviles y poder contribuir en el ahorro energético.

Uno de los obstáculos que se puede encontrar este proyecto, es el hecho que pueda haber dispositivos móviles con sensores de muy baja calidad donde alguno de los métodos no funcione del todo bien, pero estos casos son muy pocos.

Además de la comparación también se pretende que este proyecto haga de puente entre un proyecto de movilidad y otro de contaminación del grupo de investigación. Utilizando así métodos de fusión de sensores para localizar usuarios y derivar modelos de contaminación.

2 | Gestión del Proyecto

2.1. Metodología y Rigor

El método empleado en este trabajo es *Cascada*, en el que se realizan las diferentes tareas y al terminar cada una se revisa para ver si se puede mejorar o avanzar a la siguiente. Semanalmente se han hecho reuniones con el director y el grupo de investigación donde se lleva a cabo el proyecto. Para así ver los obstáculos encontrados, proponer posibles mejoras e ir validando el progreso. No se ha validado el trabajo hecho al final del proyecto, sino que se ha ido validando a medida que se han ido implementando métodos o incorporando mejoras. Así en las reuniones con los profesores del grupo de investigación se han ido comprobando los resultados para ver que todo avanzaba según lo planeado. Una vez terminadas cada una de las tareas se revisaron para ver si se podía hacer alguna mejora y así hacer otra iteración.

Se utilizan tres herramientas para dar soporte al proyecto. En primer lugar, *GitLab*, una herramienta de versiones y mantenimiento de código. En segundo lugar, el *Google Drive* para compartir documentos como por ejemplo mapas con los resultados de diferentes pruebas. Y finalmente *Trello*, una web que permite crear un tablero donde poner las diferentes tareas a hacer y poner fechas límite para terminarlas, pudiendo hacer así un seguimiento temporal.

2.2. Planificación Temporal

2.2.1. General

La duración del proyecto es de 5 meses aproximadamente, de inicios de Febrero de 2017 a finales de Junio.

2.2.2. Tareas

Entre las diferentes tareas especificadas en la tabla 2.1 (con sus estimaciones temporales iniciales y las resultantes), podemos extraer las tres fases principales del proyecto. La primera fase comprende el planteamiento del proyecto y la adquisición de datos. El planteamiento se basa en la realización del curso de GEP y la lectura de la literatura existente sobre el tema de estudio. En la adquisición de datos, se harán retoques a una aplicación ya existente de recogida de datos, se recolectarán datos en diferentes tests y se creará el código para realizar el pre-procesado de estos datos.

En la segunda fase encontramos la implementación de los dos algoritmos principales utilizados para la comparación (el sistema de navegación inercial y el filtro de Kalman).

Finalmente, la última fase comprende la comparación de los diferentes métodos implementados, la redacción de la memoria y elaboración de la presentación. Las relaciones temporales se encuentran en el gantt (ver anexo 1).

En cuanto a las dependencias, las principales dependencias se encuentran dentro de los mismos bloques, por ejemplo, para implementar el algoritmo INS primero debe haber un aprendizaje. Estas dependencias también se muestran en el gantt (ver anexo 1).

Cuadro 2.1:Tareas

Tarea	Inicial	Final
1. Planteamiento del proyecto	120	120
1.1 Definir el proyecto (GEP)	85	85
1.2 Estudio de la literatura existente	35	35
2. Adquisición de datos	55	55
2.1 Mejoras aplicación recogida datos sensores	10	10
2.2 Servidor datos sensores	5	5
2.3 Recolección de datos en trayectos	20	20
2.4 Pre-procesado datos	20	20
3. INS	100	84
3.1 Aprendizaje sobre cinemática y coordenadas	40	40
3.2 Implementación INS	20	27
3.3 Técnicas corrección error	20	0
3.4 Testeo	20	17
4. Filtro de Kalman	130	133
4.1 Aprendizaje Filtro de Kalman	50	50
4.2 Diseño filtro de Kalman	40	40
4.3 Implementación	20	20
4.4 Testeo	20	23
5. Resultados	80	75
5.1 Comparación de diferentes métodos	60	55
5.2 Conclusiones	20	20
6. Documentación	90	96
6.1 Redacción memoria	60	66
6.2 Elaboración presentación	30	30
TOTAL	575	563 h.

2.2.3. Recursos

En cuanto a recursos humanos solo está el autor del proyecto que tal y como se refleja más adelante (en la sección de costes) lleva a cabo diferentes roles. Entre los diferentes recursos hardware y software utilizados para llevar a cabo todas las tareas se encuentran:

- Portátil HP Pavilion g6: ordenador utilizado para elaborar todo el proyecto.
- Máquina virtual del Departament d'Arquitectura de Computadors: alojamiento del servidor que recoge datos de la aplicación.
- Samsung Galaxy Mini S3: teléfono utilizado para hacer los tests.
- Google Drive: almacenamiento mapas con los resultados de las pruebas.
- Trello: seguimiento de las tareas.
- GitLab: mantenimiento del código.
- SublimeText: editor de texto para escribir el código necesario.
- Android Studio: modificaciones aplicación recolección de datos.
- Ganttter: web para la elaboración del gantt.
- Overleaf: editor online LaTeX.
- Python: lenguaje de programación utilizado para implementar los algoritmos.

2.2.4. Desviaciones

Las principales desviaciones que puede haber están en la implementación del INS y el filtro de Kalman. Dichas implementaciones pueden durar más de lo previsto si se encuentran problemas, o simplemente emplear más tiempo mejorando estos métodos. Estos posibles contratiempos se han tenido en cuenta añadiendo más tiempo en las fases de testeo.

Otra desviación que se ha tenido en cuenta es el hecho que los datos recogidos en la fase de recolecta de datos no sean suficientes. Sin embargo, dado

el incremento en el tiempo de testeo, esta desviación tampoco afectaría a la terminación del proyecto a tiempo.

Finalmente de las dos posibles desviaciones se produjo la primera. La fase de testeo del filtro de Kalman se alargó un poco ya que se hicieron diferentes pruebas con diferentes valores para mejorar la solución obtenida. Pero a la vez, la fase de testeo del INS resultó ser más corta.

La implementación del INS se alargó 7 horas debido a diferentes errores en la transformación de coordenadas. La parte de técnicas de corrección de error no llevaba a ningún sitio (se intentaron regresiones lineales para quitar el error a las medidas) y se decidió no seguir adelante con esta tarea. Y la redacción de la memoria duró 6 horas más debido a la generación de múltiples gráficos que no estaban previstos.

2.3. Gestión económica

2.3.1. Estimación de Costes

En la estimación de costes no se han tenido en cuenta los costes de producción ya que que no se ha creado ningún tipo de unidad, sino que se ha desarrollado software y se ha aportado conocimientos. De este modo solo se presentan los costes de infraestructura (los únicos). A continuación se especifican los principales costes directos.

Costes Directos

Los principales gastos son los recursos humanos. Este proyecto se ha llevado a cabo por una sola persona que se ha encargado de los tres roles principales, siendo jefe de proyecto, programador y tester a la vez.

Cuadro 2.2:Distribución horas Gantt

Etapas	Jefe Proyecto	Programador	Tester	Total
1. Planteamiento proyecto	120	0	0	120
2. Adquisición Datos	20	35	0	55
3. INS	40	27	17	84
4. Filtro Kalman	90	20	23	133
5. Comparación	20	55	0	75
6. Documentación	96	0	0	96
Total	386 h	137 h	40 h	563 h

Cuadro 2.3:Costes humanos

Rol	Precio	Horas	Total
Jefe de proyecto	30	386	11580
Programador	20	137	2740
Tester	20	40	800
Total		563	15120 €

En segundo lugar encontramos los gastos por recursos hardware descritos con sus respectivas amortizaciones a 5 meses.

Cuadro 2.4:Costes hardware

Recursos Hardware	Precio	Unidades	Vida útil	Amortización
Portátil HP pavilion g6	699	1	3 años	89.61
Samsung Galaxy Mini S3	150	1	2 años	31.25
Màquina Virtual	120	1	1 año	50
Total				170.86 €

En cuanto al software, el uso de los programas especificados en la anterior sección no conlleva ningún coste ya que se trata de programas *OpenSource*, es decir, de uso gratuito y código abierto.

Otros costes

A parte de los costes directos también se han tenido en cuenta los siguientes costes:

- Indirectos: dentro de estos gastos está la luz, el gas, etc. En este caso los gastos ascienden a 520 €al mes , 500 por el alquiler de la oficina (que incluye la electricidad, calefacción, Internet,etc) y 20 por la compra de billetes de transporte público.
- Contingencia: se establece un 15 % adicional como contingencia de las posibles desviaciones que pueden ocurrir. Dichas desviaciones ya han sido vistas en el apartado de desviaciones. El importe asciende a 2268€.
- Imprevistos: se destinan 100 €a imprevistos. Entre los diferentes imprevistos posibles tenemos que se estropee el portátil o el teléfono móvil, con lo cual 100 euros podrían cubrir la reparación. La probabilidad que se estropee uno ya es pequeña con lo cual que se estropeen los dos en el mismo periodo de tiempo es mucho menor y no se tiene en cuenta.

Total Costes

Cuadro 2.5:Costes totales

Directos	Indirectos	Contingencia	Imprevistos	Total
15120	520	2268	100	18008 €

2.3.2. Control de gestión

Se han realizado reuniones semanales con el tutor del proyecto. En estas reuniones (de 45 minutos) se ha comprobado el progreso hecho durante toda la semana, y además se discutieron posibles mejoras o correcciones.

Además, para estas reuniones se ha hecho un resumen de las horas empleadas por cada uno de los roles y los recursos utilizados. Para así poder observar si se producían desviaciones de la planificación original, en el caso de haber alguna desviación se estudió la causa para evitarla en un futuro. La detección de las desviaciones es sencilla, en las reuniones semanales se observó la planificación para ver si todo seguía su curso normal.

Las principales desviaciones económicas que podían haber eran causadas por desviaciones temporales, ya que el hecho de necesitar más horas de recursos humanos afecta al presupuesto planificado. Pero dichas desviaciones ya se han nombrado en el capítulo anterior.

3 | Sostenibilidad y compromiso social

3.1. Impacto ambiental

El consumo energético hecho durante el proyecto depende directamente del consumo del portátil, el móvil y el consumo de la máquina virtual. Teniendo en cuenta que son máquinas relativamente nuevas (menos 3 años), los componentes ya están hechos para tener un bajo consumo. Además para poder recoger datos de trayectos y así realizar su posterior análisis se ha utilizado transporte público (tranvía), contaminando menos. No se ha utilizado ningún documento en soporte físico para ahorrar papel.

Actualmente otros estudios en los que se usan datos reales han sido tomados sobre un vehículo convencional (no transporte público). Además la mayoría de proyectos se centran solo en la mejora de precisión del método y no contemplan los costes energéticos. En este proyecto, en cambio, se tiene siempre en cuenta el consumo energético, intentando implementar métodos que puedan reducir el consumo de los métodos actuales.

El proyecto puede tener un importante impacto en el ahorro energético en el campo de los dispositivos móviles, ya que puede dar conocimiento de métodos que consumen menos batería que los métodos convencionales (GPS). Así, si hay dos métodos que pueden encajar en las necesidades de una aplicación, elegir el que consuma menos. Por todo esto, el uso del proyecto mejoraría la huella ecológica.

Los datos del proyecto también se podrían utilizar sin tener en cuenta el consumo energético, usando el método que tenga mejor precisión aunque consuma más. Si se usa de este modo, se corre el riesgo que el proyecto pueda afectar de forma negativa (aunque mínima) a la huella ecológica.

3.2. Impacto económico

En la estimación de costes se han tenido en cuenta costes directos (recursos humanos, hardware y software) , costes indirectos, costes de contingencia y costes por imprevistos. El tiempo estimado para cada tarea está basado en su importancia. Es posible que al terminar este proyecto los algoritmos hechos y las conclusiones sacadas se puedan utilizar para otros proyectos. El coste del proyecto está delimitado a las cosas esenciales (recursos humanos y hardware necesario), haciendo así de la reducción de costes una tarea muy difícil.

El coste previsto ha resultado ser bastante parecido al coste final. Pese haberse producido diferentes cambios en la duración de diferentes tareas, la previsión total de horas ha sido bastante buena (12 horas de diferencia).

3.3. Impacto social

Este proyecto se desarrolla en la era de las comunicaciones y de los smartphones. Aportar información para poder utilizar recursos de localización de forma más eficaz puede tener un gran impacto, desde consumir menos energía, a tener los usuarios localizados con más precisión, para así poder desarrollar aplicaciones que puedan satisfacer mejor sus necesidades.

Tal y como se verá más adelante, hay métodos con propiedades específicas que pueden ser útiles en aplicaciones críticas como las detección de accidentes, haciendo así una localización más exacta de donde se ha producido el accidente.

El uso de la información proporcionada no puede afectar negativamente a ningún segmento social, al contrario, si se consigue que las técnicas de menor consumo se utilicen más, los usuarios también se beneficiarían (reduciendo el consumo de batería de sus dispositivos).

A nivel personal el proyecto me ha hecho reflexionar sobre la necesidad de intentar, ya sea en la localización u otras áreas, conseguir el objetivo al menor coste energético posible, ya que es probable que existan dos maneras de hacer alguna cosa, pero una puede tener menor coste energético. Además, creo que el proyecto puede ayudar a contribuir al ahorro energético utilizando las conclusiones extraídas como referencia. El hecho que se puedan mejorar los métodos existentes puede que haga reflexionar a los desarrolladores e incluso

hace que utilicen el método que se adecue a sus necesidades.

Cuadro 3.1: Matriz de sostenibilidad

	PPP	Vida útil	Riesgos
Ambiental	8	15	-3
Económico	7	13	0
Social	9	12	0
Rango	24	40	-3
Sostenibilidad	61		

4 | Sistemas de Localización Outdoor

4.1. GPS

Uno de los métodos de localización en exteriores (localización outdoor) más utilizado es el Global Positioning System (*GPS*). Este sistema de geolocalización fue creado por el departamento de defensa de los EEUU con fines militares, siendo posteriormente utilizado con fines civiles de forma generalizada. Aunque este proyecto se centre en el GPS, existen otros sistemas similares, tales como el Galileo (Europa) o el GLONASS (Rusia).

El uso de este sistema permite obtener valores de posición muy precisos (± 15 metros). El sistema GPS se basa en explotar la información proporcionada por un conjunto de satélites (24 aproximadamente) en órbita alrededor de la Tierra. Está compuesto por tres capas:

- Capa de satélites: la compone el conjunto de satélites en órbita que van retransmitiendo mensajes con información sobre su posición y tiempo de transmisión. Los satélites disponen de relojes atómicos de extraordinaria precisión y exactitud, y siguen órbitas predeterminadas cuya trayectoria exacta es constantemente tenida en cuenta para que pueda incorporarse en los cálculos de geolocalización.
- Capa de control: la compone el conjunto de estaciones de monitorización que se encargan de revisar el estado de los satélites y hacer correcciones de las desviaciones de los relojes internos.
- Capa de los usuarios: compuesta por cada uno de los dispositivos con receptores *GPS*.

A grandes rasgos la operación del GPS puede resumirse así: los satélites periódicamente envían información sobre su posición y tiempo de envío del

mensaje. Los receptores *GPS* reciben estas señales y partir de esta información son capaces de determinar la posición en la que se encuentran mediante algoritmos de multilateración¹, determinando así la posición del receptor en un elipsoide de revolución. Finalmente traducen dicha posición a la posición sobre la Tierra, expresada en el sistema de coordenadas geodésico (World Geodetic System (*WGS*)), también llamado LLA (*Latitude-Longitude-Altitude*).

Hay varias fuentes de error a la hora de determinar la localización. Una de estas fuentes es el ruido de la electrónica del receptor y las imprecisiones del reloj del receptor. Otra fuente de error es la transmisión multitrayecto (*multi-path*), que consiste en la recepción de una misma señal en tiempos distintos a causa de reflexiones de la misma en otros objetos. Este error puede introducir un error de hasta 100 metros en el cálculo de la posición, pero existen métodos para atenuar estos errores. Otro error común es el causado por la refracción de la señal GPS en la troposfera. Finalmente, el bloqueo de señal debido a grandes objetos también puede ser origen de error.

En cuanto al consumo, este método de localización requiere la utilización de un receptor GPS, que incrementa el consumo del dispositivo que lo lleva integrado.

En conclusión, el GPS es un sistema de localización muy preciso pero que también conlleva un elevado consumo de potencia.

4.2. Redes Celulares

Otro sistema de localización ampliamente utilizado es la localización mediante redes celulares.

En su forma más sencilla, se basa en determinar los identificadores de las torres de telefonía celular recibidas por el teléfono, y de ahí deducir la localización. También se puede obtener la posición mediante multilateración de la información de diferentes torres o incluso dependiendo de la intensidad de señal de la célula.

Este método es menos preciso que el *GPS* ya que el error cometido está directamente relacionado con el tamaño de la célula, que dependiendo del tipo

¹Multilateración: técnica de navegación que utiliza más de una diferencia en distancia entre dos estaciones para reducir el número de posibles localizaciones.

de entorno puede ser de centenares de metros o incluso de kilómetros. Sin embargo, el uso de este tipo de localización en los smartphones no necesita de hardware adicional, sino que aprovecha la propia conexión a la red de telefonía móvil. Por eso, el consumo de éste método es mucho menor que el del GPS ya que no necesita tener activo ningún componente adicional.

4.3. Sistemas de Coordenadas Geográficas

4.3.1. Sistema LLA

El LLA (también llamado sistema geodésico) es el sistema utilizado por el GPS. En este sistema un punto está compuesto por longitud, latitud y altitud. La longitud es el ángulo (en grados) que forma el punto con el primer meridiano, la latitud mide el ángulo (en grados) entre el plano ecuatorial y la normal de la elipse de referencia que pasa por el punto. Y la altitud es la distancia vertical (en metros) entre el punto y la elipse de referencia.

4.3.2. Sistema ECEF

El sistema ECEF (*Earth-Centered Earth-Fixed*), es un sistema de coordenadas cartesiano donde el punto $(0,0,0)$ corresponde al centro de la Tierra. El eje Z corresponde al eje de la tierra apuntando hacia el polo norte, el eje X corresponde al vector que secciona la esfera de la Tierra por el punto de latitud 0° y longitud 0° . Y finalmente el eje Y que es ortogonal a los dos anteriores.

4.3.3. Sistema NED

Este sistema, *North-East-Down*, es un sistema frecuentemente utilizado en navegación aeroespacial. Es un sistema de coordenadas de navegación donde el origen de coordenadas corresponde a un punto arbitrario de la tierra, formando un plano tangente a la superficie de la Tierra en dicho punto. El eje N apunta hacia el norte geodésico, el eje E apunta hacia el este geodésico y el eje D apunta hacia la normal del plano (centro de globo terráqueo).

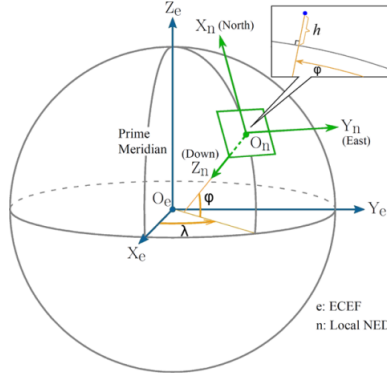


Figura 4.1: Sistemas de coordenadas ECEF y NED. Fuente: [2]

4.4. Localización en Android

Existen diferentes APIs para localizar un dispositivo Android, entre ellas está la propia API del sistema Android y la API de Google, el **Fused Location Provider**. En este proyecto se utiliza la última API ya que es la recomendada para realizar tareas de localización con smartphones.

Esta API determina la posición mediante diferentes métodos según los permisos que tenga la aplicación y la precisión requerida. Principalmente dispone de tres prioridades : *high_accuracy* , *balanced_power*, *no_power*. La primera requiere de datos de mucha precisión y por eso utiliza el hardware(GPS), que es el que puede dar más precisión. La segunda, dependiendo de lo que tiene disponible localiza el dispositivo mediante torres celulares o mediante WIFI. Finalmente, la tercera prioridad es un método pasivo donde la aplicación no recibe actualizaciones de posición sino que aprovecha las peticiones de otras aplicaciones. Hay que tener en cuenta que con esta API, en todo momento dependiendo de la precisión que necesite puede combinar más de un método, por ejemplo torres celulares y WIFI.

Junto a las actualizaciones de posición también proporciona información sobre la magnitud de la velocidad del dispositivo y la orientación de éste (calculada a partir de las posiciones GPS). Estos datos pueden ser útiles para la fusión de datos.

Cuadro 4.1: Datos FusedLocationProvider (*Posibles Métodos)

Prioridad	Intervalo	Consumo (%/hora)	Método*
HIGH_ACCURACY	5s	7.25 %	GPS
BALANCED_POWER	20s	0.6 %	Torres celulares/WIFI
NO_POWER	NA	0.0 %	Pasivo

5 | IMU

Los métodos de localización desarrollados en este proyecto se basan en *Inertial Measurement Units* (IMUs). Las IMUs son dispositivos que proporcionan información sobre las fuerzas ejercidas sobre el dispositivo, básicamente se componen de acelerómetros y giroscopios, donde éstos están fijados a la IMU (*strapdown IMU*). En este caso la IMU utilizada no es ningún dispositivo externo, sino que se utilizan los propios sensores incorporados dentro de los smartphones. El objetivo es integrar las medidas de estos sensores para poder derivar la posición y orientación del móvil.

Otro sensor utilizado en el proyecto es el magnetómetro, que mide el campo magnético en un punto (en microteslas) y permite calcular la dirección actual respecto al norte magnético (comúnmente llamada *azimut*). Más adelante se explicará cómo se incorporan las medidas de este sensor para corregir la estimación de orientación.

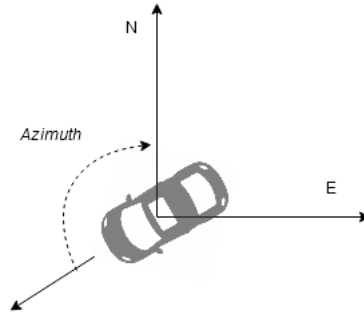


Figura 5.1:Ángulo Azimut

5.1. Sensores

La mayoría de IMUs presentes en dispositivos móviles incorporan acelerómetros y giroscopios de tres ejes ortogonales (esto significa que miden en tres direcciones ortogonales¹ entre ellas). Los acelerómetros miden la fuerza ejercida sobre la IMU, tanto la aceleración causada por la gravedad como la causada por el propio movimiento de la IMU. Los giroscopios miden la velocidad angular del dispositivo (en sistema de coordenadas body frame, explicado en la siguiente sección) respecto al sistema de coordenadas inercial. La velocidad angular en body frame corresponderá a ω y la aceleración en body frame a a .

$$\omega^b = (\omega_x^b, \omega_y^b, \omega_z^b)^T; a^b = (a_x^b, a_y^b, a_z^b)^T \quad (5.1)$$

Estos sensores sufren de errores de naturaleza distinta, entre otros están:

- Bias: *Offset* constate que afecta a la lectura del sensor.
- Random Walk: Error dinámico aleatorio en la medida del sensor.
- Desalineamiento: Error de desalineación de los ejes del sensor que dejan de ser ortogonales entre si.
- Error factor de escala: Error debido a no linealidades.

La existencia de estos errores hace que los datos proporcionados por la IMU no sean del todo precisos y que en el caso del INS, al integrar estos valores, se incorpore error en el cálculo de posición. De esto modo, por ejemplo, en el caso del acelerómetro que se integra para obtener la velocidad que a la vez también se integra para derivar la posición, como resultado se obtiene que el error introducido se acumula de forma cuadrática con el tiempo en la posición y de forma lineal en la velocidad. Además, el giroscopio también se integra para derivar la orientación del dispositivo, por lo que el error en la orientación crece linealmente respecto al tiempo.

¹Direcciones ortogonales: direcciones que forman un ángulo recto entre si.

5.2. Sistema de Coordenadas Local/body

El sistema de coordenadas local o *body frame* es el que se utiliza para referenciar la información proporcionada por los sensores. Los ejes X y Y son intercambiables pero normalmente el eje X apunta en la dirección longitudinal, el Y en la transversal y el eje Z apunta hacia la dirección vertical. En la siguiente imagen se muestra un ejemplo de sistema de coordenadas local, el que se usa en los sensores de Android.

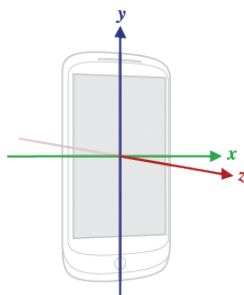


Figura 5.2:Ejes dispositivo android, donde el eje 'Y' del dispositivo corresponde al eje 'X' del sistema de referencia local. Fuente: <https://developer.android.com/reference/android/hardware/SensorEvent.html>

5.3. Sensores en Android

El SDK de Android² (Software Development Kit) proporciona una herramienta para poder leer los datos de los sensores disponibles en el móvil. Esta herramienta se llama *Sensor Manager*, y permite recibir datos de los sensores periódicamente fijado un período de muestreo.

Existen cuatro posibles períodos de muestreo, *delay_normal* (200 ms), *delay_ui* (60ms), *delay_game* (20ms), *delay_fastest* (0ms). Pese a esto, sensores como el acelerómetro y el giroscopio están activos por defecto para poder dar información como los cambios de orientación de pantalla. Por eso se ha aprovechado el echo de que ya proporcionan datos y el intervalo de muestreo seleccionado ha sido el mayor para consumir menos energía. Las unidades utilizadas por el

²SDK: conjunto de herramientas software utilizadas para crear aplicaciones Android

Sensor Manager son metros por segundo al cuadrado en el caso del acelerómetro, radianes por segundo en el caso del giroscopio y microteslas en el caso del magnetómetro.

Los sensores de un móvil utilizan el sistema de coordenadas local 5.2.

6 | Fusión de Sensores

La fusión de sensores es una técnica que combina lecturas de diferentes sensores para obtener información con menos incertidumbre que la información obtenida por cada sensor individual. Entre los diferentes ejemplos de fusión de sensores encontramos los implementados en este proyecto, GPS/INS y Redes Celulares/INS, y otros que combinan sistemas de visión con otros receptores. La fusión de sensores no solo se utiliza en el mundo de la tecnología, sino que también existe su símil en las personas, que utilizamos la información que obtenemos por diferentes receptores (tacto, ojos, oído) para estar seguros de lo que realmente vemos, es decir, para reducir la incertidumbre sobre un hecho.

Un primera técnica elemental para realizar la fusión GPS/INS, podría consistir en utilizar los sensores inerciales para ir derivando la posición y al recibir la localización del GPS actualizar la estimación a aquella posición. Sin embargo, hay que tener en cuenta que las medidas que recibimos no son exactas y contienen un error, por lo que esta posible técnica solo sería buena si todas las medidas recibidas no tuvieran ningún error. Por lo tanto, es necesario desarrollar técnicas más sofisticadas para realizar la fusión de sensores

6.1. Filtro de Kalman

El filtro de Kalman es un filtro Bayesiano¹ que utiliza herramientas probabilísticas para estimar el estado de un sistema dinámico (es decir, que varía con el tiempo) a partir de medidas ruidosas.

Se define un modelo de espacio del estado donde $x_t \in \mathbb{R}^n$ es el estado del sistema que se quiere estimar en el tiempo 't', y $y_t \in \mathbb{R}^m$ son las medidas exteriores recibidas. Para el estado, tenemos el modelo dinámico $p(x_t|u_t, x_{t-1})$ que modela la dinámica del sistema, siendo $u_t \in \mathbb{R}^l$ los controles recibidos

¹El filtro de Kalman también es conocido como estimador lineal cuadrático (*Linear Quadratic estimator*)

por el sistema. El modelo de medición $p(y_t|x_t)$ modela la distribución de las medidas dado el estado.

Los filtros Bayesianos utilizan la propiedad de independencia condicional de los estados de Markov, donde el estado en el paso 't' solo depende del estado en el paso anterior 't-1'.

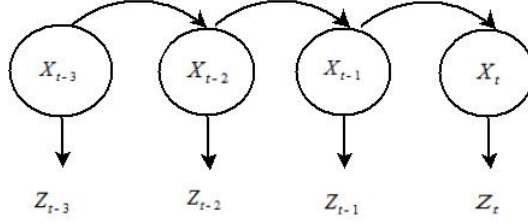


Figura 6.1: Proceso de Markov, donde tenemos los estados 'x' y las medidas ruidosas 'z' que observamos sobre los estados.

El filtro representa el posterior como una Gaussiana si se cumplen tres propiedades:

1. El modelo dinámico debe ser lineal aunque puede incorporar una perturbación modelada por un ruido blanco Gaussiano aditivo (ε). Siendo A_t la matriz de transición del modelo dinámico, B_t la matriz de introducción de los controles al estado y Q la matriz de covarianza del ruido del proceso.

$$x_t = A_t x_{t-1} + B_t u_t + \varepsilon_t; \varepsilon \sim N(0, Q) \quad (6.1)$$

$$p(x_t|x_{t-1}) = N(x_t|A_{t-1}x_{t-1} + B_t u_t, Q_{t-1}) \quad (6.2)$$

2. El modelo de medición también debe ser lineal y puede incorporar una perturbación modelada por un ruido blanco Gaussiano aditivo (δ). Siendo H_t la matriz del modelo de medición y R la matriz de covarianza del ruido de medición.

$$z_t = H_t x_{t-1} + \delta_t; \delta \sim N(0, R) \quad (6.3)$$

$$p(z_t|x_t) = N(z_t|H_t x_t, R_t) \quad (6.4)$$

3. La creencia inicial (probabilidad inicial $p(x_0)$) debe seguir una distribución Gaussiana con media m_0 y covarianza P_0 .

Con el objetivo de calcular la distribución marginal a posteriori $p(x_t|z_{1:t})$ el filtro de Kalman establece una solución cerrada al filtraje óptimo utilizando el modelo dinámico y de medición 6.1,6.3, resultando en el siguiente modelo probabilístico que sigue distribuciones Gaussianas :

$$p(x_t|z_{1:t-1}) = N(x_t|m_t^-, P_t^-) \quad (6.5)$$

$$p(x_t|z_{1:t}) = N(x_t|m_t, P_t) \quad (6.6)$$

$$p(z_t|z_{1:t-1}) = N(z_t|H_t m_t^-, S_t) \quad (6.7)$$

Para deducir los parámetros del modelo anterior se deriva el siguiente algoritmo, dividido en dos pasos. El paso de predicción, que realiza la predicción del nuevo estado (m_t^- predicción media, P_t^- predicción covarianza) con el modelo dinámico, y el paso de corrección, que introduce las medidas con el modelo de medición, para corregir el estado.

$$\left. \begin{aligned} m_t^- &= A_{t-1}m_{t-1} + B_t u_t \\ P_t^- &= A_{t-1}P_{t-1}A_{t-1}^T + Q_{t-1} \end{aligned} \right\} \text{Predicción} \quad (6.8)$$

$$\left. \begin{aligned} v_t &= z_t - H_t m_t^- \\ S_t &= H_t P_t^- H_t^T + R_t \\ K_t &= P_t^- H_t^T S_t^{-1} \\ m_t &= m_t^- + K_t v_t \\ P_t &= P_t^- - K_t S_t K_t^T \end{aligned} \right\} \text{Corrección} \quad (6.9)$$

Siendo K_t (*Kalman Gain*) el grado en el que se introducen las medidas al estado (cuanto más grandes los valores de la matriz más se introducen las medidas). Los valores de esta matriz dependen de la incertidumbre que hay en el estado y en las medidas recibidas, ponderando así las medidas que tienen más importancia para corregir el estado.

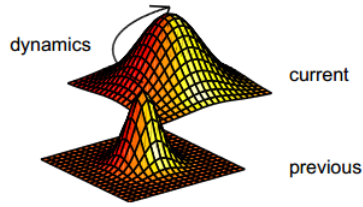


Figura 6.2: Modelo dinámico donde a partir del estado anterior y los controles ($'u'$) se deriva la predicción del estado actual habiendo aumentado la incertidumbre a causa del ruido de sistema. Fuente: [18]

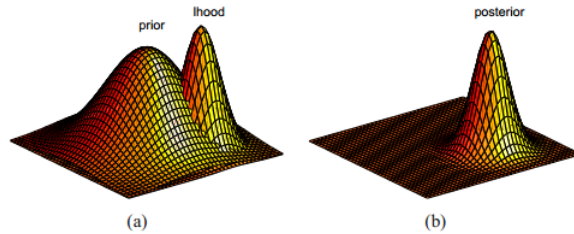


Figura 6.3: Paso de corrección donde en la primera figura tenemos la predicción del estado con más incertidumbre y la distribución de las medidas con menos. En la corrección, con la combinación de Gaussianas obtenemos otra Gaussiana. Fuente: [18]

6.2. Filtro de Kalman Extendido

En muchos casos, las ecuaciones del modelo dinámico o de medición son complejas y resultan ser no lineales, por lo que la anterior metodología no puede ser aplicada. En este caso, el filtro de Kalman extendido utiliza expansiones de las series de Taylor para linearizar el modelo alrededor de un punto (media), resultando en aproximaciones de Gaussianas. Y el siguiente modelo dinámico y de medición, donde $f(x)$ es la función del modelo dinámico y $h(x)$ es la función del modelo de medición.

$$x_t = f(x_{t-1}, u_t) + q_{t-1} \quad (6.10)$$

$$z_t = h(x_t) + r_t \quad (6.11)$$

El nuevo algoritmo también tiene el paso de predicción y corrección.

$$\left. \begin{aligned} m_t^- &= f(m_{t-1}) \\ P_t^- &= F_x(m_{t-1}) P_{t-1} F_x^T(m_{t-1}) + Q_{t-1} \end{aligned} \right\} \text{Predicción} \quad (6.12)$$

$$\left. \begin{aligned} v_t &= z_t - h(m_t^-) \\ S_t &= H_x(m_t^-) P_t^- H_x^T(m_t^-) + R_t \\ K_t &= P_t^- H_x^T(m_t^-) S_t^{-1} \\ m_t &= m_t^- + K_t v_t \\ P_t &= P_t^- - K_t S_t K_t^T \end{aligned} \right\} \text{Corrección} \quad (6.13)$$

Donde $H_x(m_t)$ y $F_x(m_t)$ son las matrices jacobianas de 'f' y 'h' respecto al estado.

$$\begin{aligned} [F_x(m)] &= \left. \frac{\partial f_i(x)}{\partial x_j} \right|_{x=m} \\ [H_x(m)] &= \left. \frac{\partial h_i(x)}{\partial x_j} \right|_{x=m} \end{aligned}$$

Debido al hecho que el filtro de Kalman extendido hace uso de la linealización y aproximación a Gaussianas, el resultado deja de ser óptimo.

7 | Implementación: Recolección de datos

Para poder probar y hacer la comparación de los diferentes métodos primero hay que disponer de datos reales. Para poder recoger los datos de los sensores, el GPS y las torres celulares, se ha utilizado una aplicación móvil hecha para el grupo de investigación. A dicha aplicación se le ha modificado la recogida de datos del GPS para que use el *FusedLocationProvider*, se ha instrumentado el código para poder localizar mediante las torres celulares (red celular), se ha creado un pequeño servidor de csv para no solo guardar los datos recogidos en local sino también en remoto. Y finalmente se le ha añadido código para hacer una evaluación de consumo energético.

Tanto la implementación del pre-procesado como la implementación de los diferentes algoritmos se ha realizado en *Python 2.7*¹, dado que proporciona librerías que facilitan las operaciones con matrices.

7.1. Pre-procesado de datos

Una vez recogidos los datos, la aceleración se debe filtrar porque los acelerómetros que incorporan las IMUs miden la fuerza que ejerce el dispositivo para contrarrestar la fuerza de la gravedad. Por ejemplo, en estático la componente vertical mide $+9,8m/s^2$ (fuerza contraria para contrarrestar la gravedad). Por eso, a la más ligera inclinación la gravedad afecta a las lecturas de los demás ejes del acelerómetro. No existe un método único y completamente satisfactorio para eliminar el efecto de la gravedad de las lecturas del acelerómetro, pero en este caso, al tener el dispositivo móvil en una posición fija (horizontal respecto al suelo) el filtrado es viable. También es ventajoso filtrar la aceleración para que introduzca menos error al ser integrada para encontrar la velocidad y la

¹Python: lenguaje de programación interpretado y que usa tipado dinámico. Muy usado actualmente.

posición.

El filtro seleccionado ha sido un sencillo filtro paso-alto de primer orden. Primero se aplica un filtro paso-bajo para aislar las bajas frecuencias (gravedad) y luego se resta la gravedad a la salida del sensor para obtener las frecuencias altas (aceleración IMU). Se puede utilizar porque al tener el móvil en una posición concreta el vector de gravedad no puede cambiar de manera brusca y tener componentes de alta frecuencia.

La siguiente formula corresponde al filtro utilizado, la selección de la α se hizo mediante prueba y error, utilizando la que filtra mejor según las necesidades ($\alpha = 0,97$).

$$gravedad_t = \alpha \cdot gravedad_{t-1} + (1 - \alpha) \cdot aceleración_t \quad (7.1)$$

$$aceleración_t = aceleración_t - gravedad_t \quad (7.2)$$

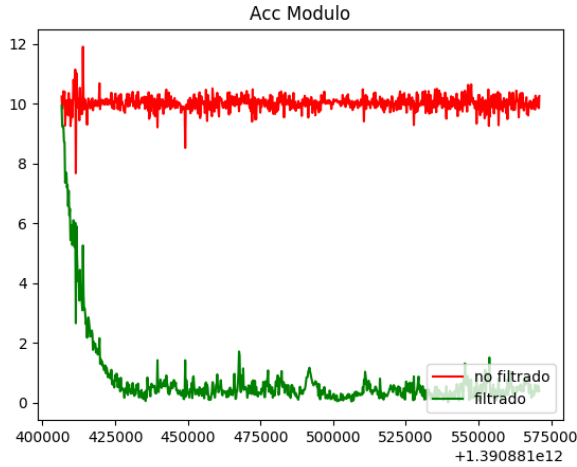


Figura 7.1:Filtrado módulo acelerómetro con un $\alpha=0.97$

A parte del pre-procesado del acelerómetro también se tiene que calcular el ángulo *azimut* (en grados) a partir de las lecturas de campo magnético del magnetómetro. Solo son necesarios los valores de los ejes 'x' y 'y'.

$$azimut = \left(\frac{180,0}{\pi} \cdot atan\left(\frac{y}{x}\right) \right) \% 360,0 \quad (7.3)$$

En los siguientes gráficos se puede ver un ejemplo de unos datos recogidos mediante la aplicación. Tal y como se puede observar, los sensores caracterizan los movimientos realizados por el vehículo, identificando paradas y cambios de dirección.

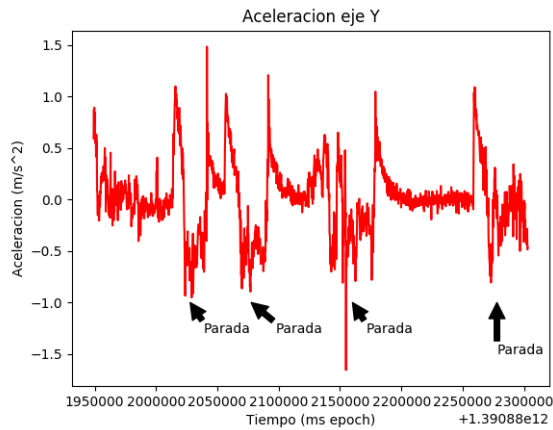


Figura 7.2:Ejemplo datos acelerómetro eje longitudinal

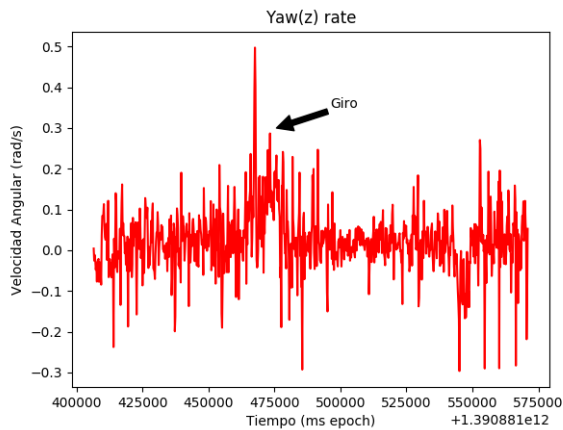


Figura 7.3:Ejemplo datos giroscopio

8 | Implementación: Modelo INS

El sistema navegación inercial (INS) se basa en las ecuaciones de cinemática, que indican como se integran los sensores de la IMU para poder describir la trayectoria de un objeto. Para poder integrar las medidas se debe utilizar el mismo sistema de coordenadas, el sistema de coordenadas seleccionado para guardar la información del estado ha sido el NED (muy utilizado en aeronaves). Por eso, es muy importante transformar los datos recibidos por la IMU que están en el sistema body frame a el sistema NED. La transformación de coordenadas se lleva a cabo en el llamado bloque de mecanización.

8.1. Ecuaciones de cinemática

Para integrar las medidas de los sensores se han utilizado simples ecuaciones de cinemática, que consisten en la integración en el tiempo de las medidas de los sensores. No se ha tenido en cuenta el efecto de la gravedad ya que los componentes de la gravedad se eliminan en el pre-procesado de señal. Tampoco se ha tenido en cuenta el efecto de la aceleración de Coriolis¹, se hubiese tenido en cuenta si los trayectos tuvieran que ser muy largos (en navegación aérea se utiliza). También se asume que la aceleración es constante durante el intervalo de muestreo, ya que en un intervalo pequeño ésta puede variar poco. El modelo

¹Efecto aceleración de Coriolis: describe como la rotación de la Tierra afecta a las medidas del giroscopio y acelerómetro

resultante es el siguiente:

$$P_f = P_i + \int v dt + \int \int a dt \xrightarrow{\text{Discreto}} P_f = P_i + T \cdot V_i + \frac{T^2}{2} \cdot a \quad (8.1)$$

$$V_f = V_i + \int a dt \xrightarrow{\text{Discreto}} V_f = V_i + T \cdot a \quad (8.2)$$

$$\psi_f = \psi_i + \int \omega dt \xrightarrow{\text{Discreto}} \psi_f = \psi_i + T \cdot \omega \quad (8.3)$$

Siendo P, V, ψ posición en NED , velocidad en NED y dirección (ángulo respecto el norte,azimut). Y a, ω aceleración y velocidad angular en NED.

El giroscopio mide la rotación de los tres ejes (pitch, roll, yaw)^{8.1}. Mediante ángulos de Euler², dado un ángulo de rotación respecto a un eje, se puede hacer una rotación de los ejes del sistema local del dispositivo (transformando así el sistema de coordenadas). En el cálculo de la dirección solo se ha tenido en cuenta un eje de giro, simplificando así el modelo. El angulo de rotación más interesante es el *yaw* ya que su dirección es la vertical, siendo la que describe mejor los cambios de dirección de un dispositivo montado en un vehículo, y los otros son insignificantes.

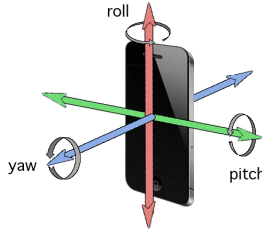


Figura 8.1:Ejes giroscopio.Fuente:<https://i.stack.imgur.com/JZfTc.png>

²Ángulos de Euler: conjunto de tres ángulos que especifican orientación de un sistema de referencia.

8.2. Transformación sistema de coordenadas

La transformación de coordenadas es muy importante para integrar los sensores, pues las medidas recibidas son relativas a la dirección a la que apunta el dispositivo. Sin embargo, la información global (posición y velocidad) se guarda en un sistema de coordenadas de navegación (NED). Como bien se ha explicado en la sección de fundamentos, el sistema de coordenadas NED consiste en tres ejes, uno apuntando hacia el norte, otro al este y el último hacia abajo.

Manteniendo el ángulo de Euler azimut (ángulo entre la dirección del dispositivo y el norte, obtenido con el eje de giro yaw), la transformación para convertir un vector de **body frame** a **NED** se hace mediante la matriz de rotación C_b^n .

$$a^n = C_b^n a^b \quad (8.4)$$

$$C_b^n = \begin{bmatrix} \cos \psi & \sin \psi & 0 \\ -\sin \psi & \cos \psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (8.5)$$

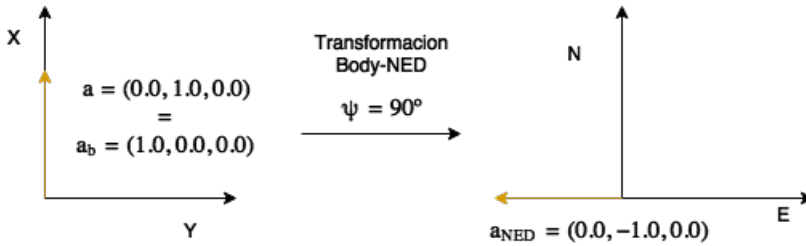


Figura 8.2: Transformación vector aceleración (en amarillo) de body frame (con una dirección de 90 grados respecto al norte) a NED.

Al no tener en cuenta las rotaciones pitch y roll, la velocidad angular (yaw) no necesita de transformación de coordenadas:

$$\omega^n = \omega^b \quad (8.6)$$

También hay que tener en cuenta que el cambio de dirección no sólo afecta a la conversión de coordenadas, sino que también afecta a la velocidad. En

concreto, al vector de la velocidad se le debe cambiar la dirección según la rotación hecha, de este modo, la velocidad inicial utilizada en la ecuación 8.2 es transformada según el incremento del ángulo de la dirección. Para realizar el cambio de dirección se puede utilizar la matriz de rotación vista 8.5, pero en lugar de utilizar el azimut se utiliza el incremento del ángulo azimut. Al ser valores muy pequeños dada la pequeña magnitud de la velocidad angular y del incremento de tiempo, se puede utilizar la aproximación de ángulos pequeños donde el coseno y el seno de un valor muy pequeño se aproximan a uno y al mismo valor respectivamente:

$$C(\Delta t \cdot \omega) = \begin{bmatrix} \cos(\Delta t \cdot \omega) & \sin(\Delta t \cdot \omega) & 0 \\ -\sin(\Delta t \cdot \omega) & \cos(\Delta t \cdot \omega) & 0 \\ 0 & 0 & 1 \end{bmatrix} \approx \begin{bmatrix} 1 & (\Delta t \cdot \omega) & 0 \\ -(\Delta t \cdot \omega) & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (8.7)$$

$$V_{NED} = C(\Delta t \cdot \omega) \cdot V_{NED} \quad (8.8)$$

Otra transformación muy importante es la conversión de **coordenadas geodésicas a NED**, para poder integrar las lecturas del GPS (en la fusión de sensores). Esta conversión se realiza mediante dos pasos: la transformación de sistema geodésico a sistema ECEF y la transformación de ECEF a NED.

La conversión de geodésico a ECEF se realiza de la siguiente forma, siendo e la excentricidad de la Tierra, R el semieje mayor de la Tierra en metros y N_e el radio de curvatura del primer vertical.

$$e = 0,081819190842622$$

$$R = 6378137$$

$$N_e = \frac{R}{\sqrt{1 - e^2 \sin^2(lat)}}$$

$$\begin{bmatrix} X_{ECEF} \\ Y_{ECEF} \\ Z_{ECEF} \end{bmatrix} = \begin{bmatrix} (N_e + alt) \cdot \cos(lat) \cdot \cos(lon) \\ (N_e + alt) \cdot \cos(lat) \cdot \sin(lon) \\ ((1 - e^2) \cdot N_e + alt) \cdot \sin(lat) \end{bmatrix}$$

Luego, la transformación de ECEF a NED se realiza mediante el siguiente producto. Donde 'lat' y 'lon' son la latitud y longitud del punto donde se ha iniciado la navegación, y (x_0, y_0, z_0) es el mismo punto inicial en ECEF.

$$\begin{bmatrix} P_N \\ P_E \\ P_D \end{bmatrix} = \begin{bmatrix} -\sin(lat) \cdot \cos(lon) & -\sin(lat) \cdot \sin(lon) & \cos(lat) \\ -\sin(lon) & \cos(lon) & 0 \\ -\cos(lat) \cdot \cos(lon) & -\cos(lat) \cdot \sin(lon) & -\sin(lat) \end{bmatrix} \cdot \begin{bmatrix} x - x_0 \\ y - y_0 \\ z - z_0 \end{bmatrix} \quad (8.9)$$

9 | Implementación: Filtro de Kalman

9.1. Estructura

Existen diferentes modos de integración del GPS con el INS:

- **Loosely-coupled:** utiliza las medidas de posición (en LLA) proporcionadas por el GPS.
- **Tight-coupled:** utiliza valores no procesados del GPS, tales como el *pseudorange*¹ (haciendo más difícil su integración).
- **Ultra-tight:** la arquitectura del receptor GPS es diferente, el INS deja de ser independiente del GPS y ayuda a deducir la señal Doppler (que permite deducir la distancia entre satélite y receptor).

La integración implementada en este proyecto es la *loosely-coupled*, ya que es la más utilizada y la que encaja mejor con los datos que se pueden obtener con el GPS de los teléfonos móviles.

Además también existen dos modos de fusión de datos mediante el filtro de Kalman:

- **Directo:** El filtro de Kalman directo realiza la estimación de la posición, velocidad, etc. Para realizar dicha implementación, las ecuaciones de cinemática se integran en el filtro, resultando en sistemas no lineales en muchas ocasiones. El principal inconveniente que tiene es que si el filtro falla ya no se puede dar una solución.

¹Pseudorange: pseudodistancia entre un satélite y un receptor GPS

- **Indirecto:** En el Kalman indirecto las ecuaciones de cinemática (INS) están separadas del filtro y el filtro estima el error en la posición, en la velocidad, etc. De este modo, las observaciones del filtro consisten en la diferencia entre los valores recibidos por los receptores externos y la solución del INS, y la salida resultante del filtro sirve para corregir el INS. La principal ventaja consiste en la posibilidad de utilizar un modelo lineal y la separación entre INS y el filtro.

Para este proyecto se ha utilizado un filtro de **Kalman directo loosely-coupled**, puesto que utilizando un modelo simplificado (1 eje de rotación), el sistema resultante es lineal y no tiene sentido que el filtro pueda fallar en el caso de una aplicación, porque si la localización por GPS no funciona, se itera sobre el paso de predicción y para que fallase el filtro tendría que fallar toda la aplicación (con lo cual, el otro modelo tampoco serviría). Además, el modelo directo nos permite utilizarlo tanto en la fusión GPS/INS como en la fusión Red Celular/INS con pequeños cambios.

La estructura de integración se muestra en la siguiente figura 9.1. Por un lado está la IMU que proporciona las medidas del acelerómetro y giroscopio (en body frame), y por otro lado tenemos el GPS y el magnetómetro que proporcionan medidas para el paso de corrección del filtro. A continuación está el bloque de mecanización, que utilizando la dirección estimada por el filtro transforma los valores del acelerómetro y del GPS al sistema de navegación NED. Finalmente, el filtro de Kalman integra los datos y proporciona una estimación de la posición, de la velocidad y de la dirección azimut.

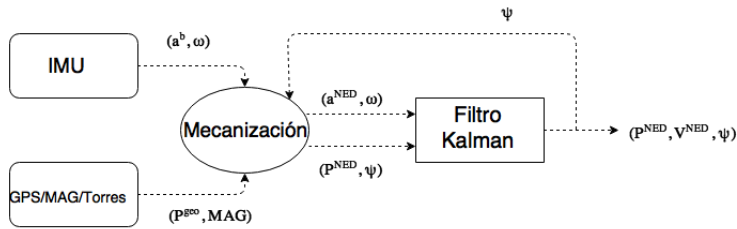


Figura 9.1: Estructura de la integración.

Hay que tener en cuenta que la frecuencia de muestreo del GPS o del método por red celular siempre va a ser menor que la frecuencia de muestreo de los sensores inerciales. Por lo tanto, mientras no hayan observaciones, se iterará sobre el paso de predicción del filtro, hasta recibir las observaciones externas.

En la siguiente imagen 9.2 se puede observar la secuencia de ejecución del algoritmo. Con los datos recibidos de la IMU se realiza el paso de predicción, donde se hace una predicción de la media y la covarianza del estado. Luego, si hay observaciones exteriores disponibles, se pasa al paso de corrección donde se corrigen las predicciones hechas en el paso anterior, pero en el caso de no haber nuevas observaciones se itera sobre el paso de predicción (con nuevos valores recibidos de la IMU).

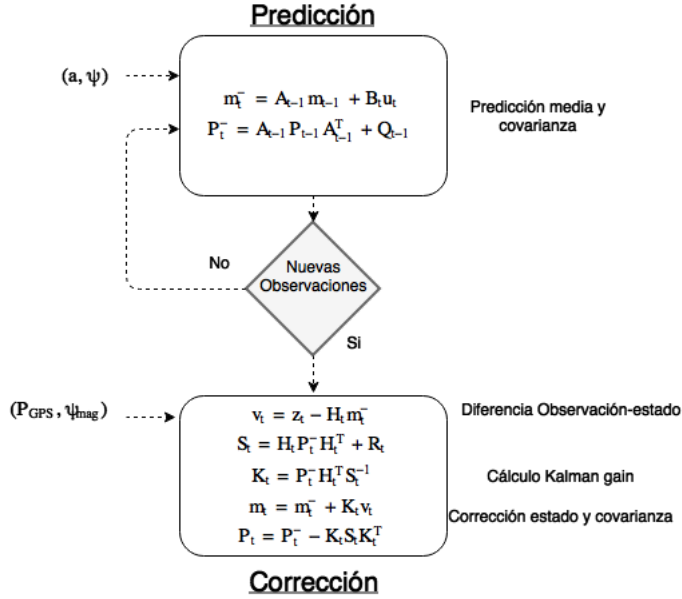


Figura 9.2: Diagrama de flujo del Kalman

Para utilizar el filtro de Kalman no solo se deben conocer los fundamentos matemáticos, sino que también se deben seleccionar los parámetros correctamente. Los parámetros son: el estado, los modelos (dinámico/medición) y las covarianzas (Q y R) del ruido de sistema y medición. Una buena elección de estos parámetros hace que el filtro sea óptimo.

9.2. GPS/INS

El estado del filtro diseñado para la integración GPS/INS se compone por la posición en NED, la velocidad en NED y el azimut. El vector de medidas (z) se compone por la posición obtenida por el GPS en NED y el azimut obtenido con los valores del magnetómetro. Finalmente, el vector de entrada o de control (u) que contiene la aceleración en NED y la velocidad angular.

$$x = (P_N, P_E, P_D, V_N, V_E, V_D, \psi)^T \quad (9.1)$$

$$z = (P_N^{GPS}, P_E^{GPS}, P_D^{GPS}, \psi^{MAG})^T \quad (9.2)$$

$$u = (a_N, a_E, a_D, \omega_{yaw})^T \quad (9.3)$$

En el bloque de mecanización 9.1 anterior al filtro se realizan las transformaciones de coordenadas. Los valores del acelerómetro se transforman de body frame a NED y los valores del GPS de sistema geodésico a NED (con las transformaciones vistas en la sección anterior).

En el modelo dinámico seleccionado se han aplicado las ecuaciones de cinemática utilizadas por el INS (visto en 8.1), de este modo, las matrices de transición y control (A y B) implementan las ecuaciones de cinemática. Y en el modelo de medición seleccionado, la matriz de observación H consiste un simple mapeo del estado al vector de observaciones.

$$x_t = \underbrace{\begin{bmatrix} 1 & 0 & 0 & \Delta t & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta t & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & 1 & \Delta t \omega & 0 & 0 \\ 0 & 0 & 0 & -\Delta t \omega & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_A x_{t-1} + \underbrace{\begin{bmatrix} \frac{\Delta t^2}{2} & 0 & 0 & 0 \\ 0 & \frac{\Delta t^2}{2} & 0 & 0 \\ 0 & 0 & \frac{\Delta t^2}{2} & 0 \\ \Delta t & 0 & 0 & 0 \\ 0 & \Delta t & 0 & 0 \\ 0 & 0 & \Delta t & 0 \\ 0 & 0 & 0 & \Delta t \end{bmatrix}}_B u_t + \varepsilon_t \quad (9.4)$$

$$z_t = \underbrace{\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_H x_t + \delta_t \quad (9.5)$$

Las siguientes matrices de covarianza del ruido de sistema (Q) y del ruido medición (R) son las seleccionadas. Para obtener los valores de dichas matrices se utilizan datos obtenidos en un período de calibración, en el cual se deja el smartphone en modo estacionario durante unos minutos. Este proceso puede ser fácilmente implementado en una aplicación, que en su primer uso se hace la calibración para obtener los valores de las matrices. Se puede observar que las matrices son simétricas y la existencia de una correlación entre el ruido en la posición y la velocidad, dado que el error en la posición también depende del error en la aceleración. En la literatura [17] aparecen matrices de covarianza similares .

$$Q = \begin{bmatrix} \frac{\Delta t^4}{4} \sigma_{accN}^2 & 0 & 0 & \frac{\Delta t^3}{2} \sigma_{accN}^2 & 0 & 0 & 0 \\ 0 & \frac{\Delta t^4}{4} \sigma_{accE}^2 & 0 & 0 & \frac{\Delta t^3}{2} \sigma_{accE}^2 & 0 & 0 \\ 0 & 0 & \frac{\Delta t^4}{4} \sigma_{accD}^2 & 0 & 0 & \frac{\Delta t^3}{2} \sigma_{accD}^2 & 0 \\ \frac{\Delta t^3}{2} \sigma_{accN}^2 & 0 & 0 & \Delta t^2 \sigma_{accN}^2 & 0 & 0 & 0 \\ 0 & \frac{\Delta t^3}{2} \sigma_{accE}^2 & 0 & 0 & \Delta t^2 \sigma_{accE}^2 & 0 & 0 \\ 0 & 0 & \frac{\Delta t^3}{2} \sigma_{accD}^2 & 0 & 0 & \Delta t^2 \sigma_{accD}^2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \Delta t^2 \sigma_{gyr}^2 \end{bmatrix} \quad (9.6)$$

$$R = \begin{bmatrix} var(GPS)^N & 0 & 0 & 0 \\ 0 & var(GPS)^E & 0 & 0 \\ 0 & 0 & var(GPS)^D & 0 \\ 0 & 0 & 0 & var(Mag) \end{bmatrix} \quad (9.7)$$

Se probaron diferentes valores de varianza y desviación, pero al final el que dio mejor resultado fue el valor obtenido del siguiente modo: $\sigma^2 = \frac{\sum x^2}{N}$, donde x son los valores de la calibración y N el número de muestras. En el caso de los valores de la covarianza del ruido de medición, los valores son la varianza de los datos recogidos por el GPS y por el magnetómetro (azimut) durante el período de calibración.

En cuanto a la inicialización del filtro de Kalman, se necesita una posición inicial obtenida por el GPS y la dirección inicial obtenida por el magnetómetro. La velocidad inicial es cero, puesto que los trayectos comienzan desde estado estacionario. Así, la matriz de covarianza del estado inicial (P_0) representa la incertidumbre de los valores de inicialización, donde la incertidumbre en posición y dirección depende de la varianza del GPS y del magnetómetro.

$$P_0 = \begin{bmatrix} \text{var}(GPS)^N & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & \text{var}(GPS)^E & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \text{var}(GPS)^D & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & \text{var}(Mag) \end{bmatrix} \quad (9.8)$$

Para ver el efecto de la selección de las covarianzas, a continuación hay un ejemplo de matriz Kalman Gain (extraída de una prueba). Dadas las matrices de covarianza anteriores y un periodo de muestreo del GPS de cinco segundos, los valores de la matriz están alrededor de los siguientes:

$$KG = \begin{bmatrix} 0,6475916 & 0,00259798 & 0,0 & 0,0 \\ 0,00259798 & 0,67335855 & 0,0 & 0,0 \\ 0,0 & 0,0 & 0,94811323 & 0,0 \\ 0,03750711 & -0,00670908 & 0,0 & 0,0 \\ 0,00619087 & 0,04100299 & 0,0 & 0,0 \\ 0,0 & 0,0 & 0,10036224 & 0,0 \\ 0,0 & 0,0 & 0,0 & 0,28704648 \end{bmatrix} \quad (9.9)$$

La Kalman Gain se multiplica por el vector diferencia entre las medidas externas y la predicción del filtro. Se puede observar como se introduce alrededor de un 65 % la diferencia en la posición norte y este, dado que el GPS introduce menos error que los sensores inerciales. En el caso del magnetómetro, que corresponde a la última fila y columna, la dirección se introduce sólo un 28 % porque en el smartphone de prueba el magnetómetro resultó ser bastante ruidoso.

También se puede observar el efecto del ruido introducido por los sensores inerciales y el efecto de la corrección del filtro, mostrando la incertidumbre de la estimación del estado en forma de elipses de error². La incertidumbre en la posición va aumentando en los pasos de predicción (a causa del error de los sensores inerciales) y en el paso de corrección se reduce, ya que las medidas del GPS contienen menos incertidumbre que la predicción del estado.

²Elipse de error: método gráfico para representar incertidumbre (ver Apéndice A)

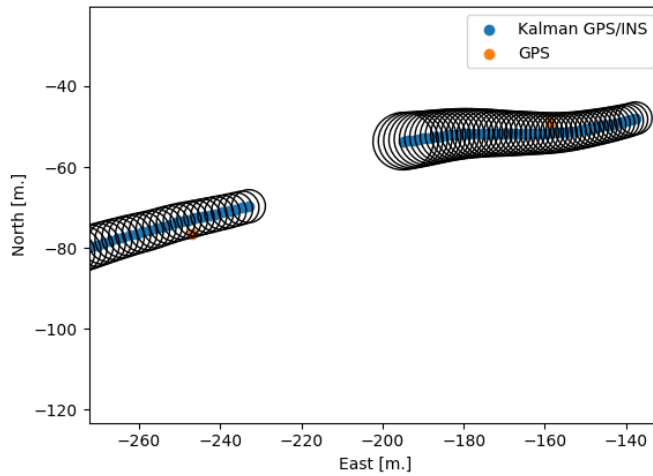


Figura 9.3: Ejemplo gráfico incertidumbre introducida por los sensores.

Tal y como se ha explicado en el capítulo de localización con Android, la API utilizada también proporciona la magnitud de la velocidad. Se probó utilizar un Extended Kalman Filter para poder corregir la velocidad a parte de la posición y dirección, pero la mejora es ínfima e interesa mantener el modelo lo más simple (lineal) posible.

9.3. Red Celular/INS

En el caso de la fusión de la localización mediante la red celular con el INS, el diseño es el mismo, lo único que cambia es la matriz de covarianza del ruido de medición y el vector de medidas (z). En este caso, en el vector de medición, la posición proviene de la localización mediante la red celular.

$$z = (P_N^{Torres}, P_E^{Torres}, P_D^{Torres}, \psi^{MAG})^T \quad (9.10)$$

El GPS reduce mucho la incertidumbre (es muy preciso), pero en el caso de utilizar la localización mediante red celular, ésta tiene mucho más error (de centenares de metros a kilómetros), con lo cual es mejor confiar más en la salida del INS. Por eso en la matriz de covarianza del ruido de medición se debe reflejar al baja precisión de este método.

$$R = \begin{bmatrix} 1500 & 0 & 0 & 0 \\ 0 & 1500 & 0 & 0 \\ 0 & 0 & 1500 & 0 \\ 0 & 0 & 0 & var(Mag) \end{bmatrix} \quad (9.11)$$

Se ha elegido una valor de varianza de las medidas igual a 1500 porque es un valor alto y así da más importancia a los sensores inerciales. Este valor podría ser un poco más alto o más bajo pero el efecto sería el mismo.

Para poder dar más precisión en este caso, el filtro se inicializa igual que el filtro anterior, con una muestra de GPS y magnetómetro. Luego pasa a recibir muestras mediante la red celular. Así, al principio el algoritmo dará más importancia a la solución del INS que a las torres celulares, pero con el paso del tiempo, el peso de la torres irá incrementando. Teniendo en cuenta esto, una posible mejora sería recibir una posición con el GPS en intervalos muy grandes para que la solución del filtro no degenera.

10 | Tests

Para poder probar y comparar los diferentes métodos implementados se recogieron datos mediante la aplicación móvil explicada en 7. Para recoger datos de un trayecto específico sabiendo cual es la trayectoria real en cada momento, se decidió utilizar el tranvía, ya que se conoce su trayectoria real. De este modo, se realizaron diferentes trayectos en repetidas ocasiones.

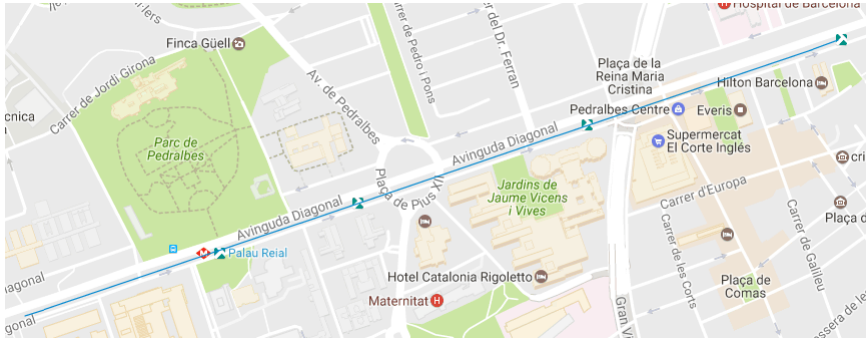


Figura 10.1: Uno de los trayectos realizados (línea verde)

10.1. GPS/INS

Se hicieron diferentes pruebas del filtro de Kalman en el caso de la fusión del GPS con el INS. En general se ve como las muestras de GPS son muy precisas, con un error aproximado de quince metros en el momento de la recepción de la posición. Se utilizó una frecuencia base de 0.2Hz para el GPS y la mínima para el INS para tener en cuenta el consumo energético.

Los siguientes mapas corresponden a la estimación del trayecto anterior 10.1. Por un lado está la estimación del sistema de navegación inercial (sin utilizar ningún tipo de corrección), pudiéndose observar como el error va creciendo

con el tiempo.

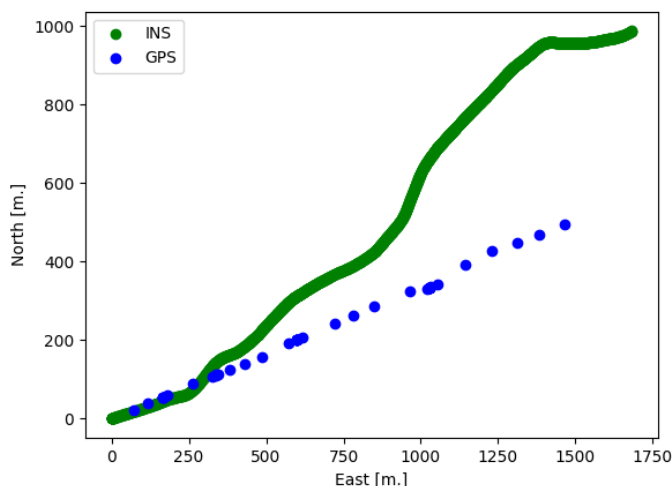


Figura 10.2: Solución INS

Por otro lado, en la gráfica del filtro de Kalman se ve como el GPS (naranja) es muy preciso. En azul está la salida del filtro de Kalman, donde se puede observar el efecto de la corrección realizada por cada muestra del GPS (la estimación de la posición cambia hacia el punto recibido). Principalmente, se aprecia como el filtro de Kalman mejora mucho la estimación del INS, pero por otro lado introduce un poco más de error que el GPS. Pese a eso, vemos que la solución del filtro de Kalman al utilizar sensores inerciales a una frecuencia más alta que la del GPS, da una información más continua sobre la posición.

También se puede observar que cuando el filtro de Kalman da peor resultado es en los momentos donde el tranvía estaba parado, puesto que todavía mantiene una velocidad residual, probablemente presente a causa de que el filtro no ha podido corregir del todo los errores de los sensores.

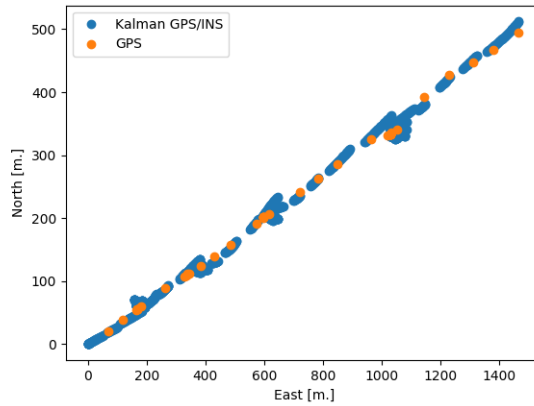


Figura 10.3: Solución fusión de sensores

A continuación, en este otro trayecto, se puede observar como el filtro da más información que el último punto recogido por el GPS. El principal problema del GPS es que durante el intervalo de muestreo, la posición estimada es la última recogida. Y en caso de la fusión hay más información (sensores inerciales), dando así una información más continua (ausente en el GPS).



Figura 10.4: Trozo trayecto entre la parada de Palau Real y Xile

Se puede ver como la estimación del filtro (puntos azules) da una visión del

movimiento realmente realizado por el tranvía (línea verde). Al contrario, el GPS (puntos rojos) simplemente da un punto antes y después de la curva.

En la siguiente gráfica se muestra el error medio en la navegación de los trayectos realizados con diferentes períodos de muestreo del GPS. El error de navegación en el caso de el GPS consiste en la diferencia entre la posición real y la última recibida por el GPS (la interpolación¹ de puntos solo se puede hacer a posteriori, en tiempo real no). Y en el caso del Kalman consiste en la diferencia entre la última estimación del filtro y la posición real. Se puede observar como a medida que aumentamos el período de muestreo del GPS, la solución del Kalman pasa a ser mejor que la del GPS. Sin embargo, con el período de muestreo más bajo (5 seg.), el GPS realiza un poco menos de error, ya que el GPS proporciona bastantes medidas.

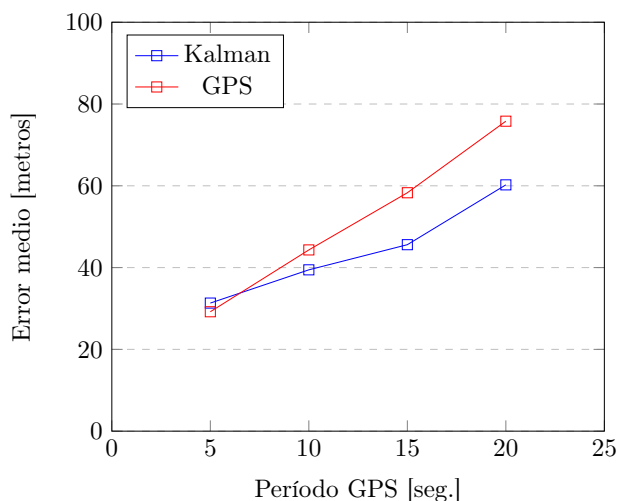


Figura 10.5

Para poder probar más propiedades de la integración utilizando el filtro, se probaron dos situaciones de manera artificial. La primera, es la introducción de un error en una de las posiciones obtenidas por el GPS (cosa que puede ocurrir debido a errores como el *multipath*) y la otra es la pérdida de señal de GPS durante unos segundos.

¹La interpolación de los puntos GPS en el caso más sencillo consiste en la unión de éstos para ver el trayecto entre los dos

En el siguiente mapa 10.6 se ha introducido un error de GPS en el trayecto (segundo punto empezando por la izquierda). En este caso, el filtro de Kalman no se come todo el error introducido por la medida. Aquí se puede observar como éste método realiza una fusión más inteligente que una fusión donde se cogería el punto GPS como correcto.

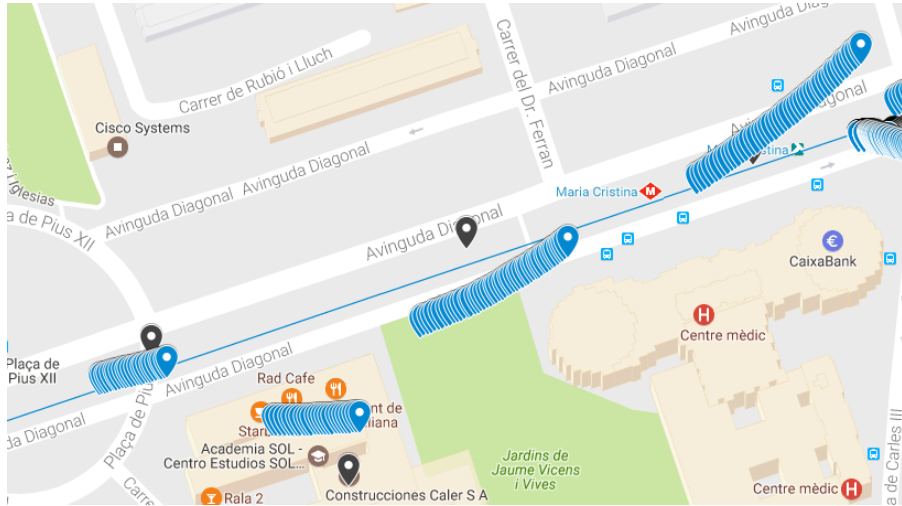


Figura 10.6:Error introducido en un trayecto (en negro los puntos GPS)

El segundo mapa 10.7 es todavía más interesante ya que simula un escenario muy común, la pérdida de señal GPS (por ejemplo, entrada en un túnel). Para realizar esta simulación no se utilizaron 5 muestras seguidas recibidas por el GPS. De este modo, se saltaron cinco pasos de corrección del algoritmo. Mediante los sensores inerciales (sin realizar ninguna corrección) se sigue estimado la posición, con error a causa de dichos sensores, pero obteniendo una solución mucho mejor que el último punto GPS recibido antes de perder la señal.

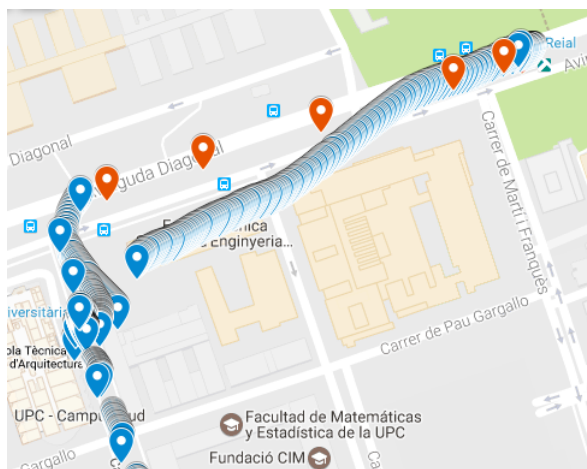


Figura 10.7: Pérdida de señal GPS durante 30 segundos. En naranja los puntos no procesados.

10.2. Red Celular/INS

La localización mediante red celular da una localización con muy poca precisión, pero es utilizada por muchas aplicaciones donde el consumo energético es crítico y solo con saber la zona donde está el usuario es suficiente.

Las pruebas realizadas con este método son muy interesantes, ya que pese a ser un método que no ofrece mucha precisión, la fusión mejora la precisión de la localización a coste mínimo.

En el siguiente mapa 10.8 se pueden observar en azul los puntos dados por las torres, donde durante casi todo el trayecto se obtuvo el punto de la esquina superior derecha, en verde la trayectoria real y en morado el Kalman. Durante el primer trozo de trayecto, la estimación no se ve alterada por la posición de la torre celular, esto se debe a la inicialización del filtro con el GPS. En los primeros pasos de corrección las medidas exteriores casi no se incorporan, dado que la solución tiene menos incertidumbre que las observaciones. A partir de ese momento, se va corrigiendo la trayectoria de modo que como más avanza en el tiempo más se corrige. Al final, la precisión obtenida mediante la fusión es mucho mejor que la del uso individual de la localización mediante la red

celular.

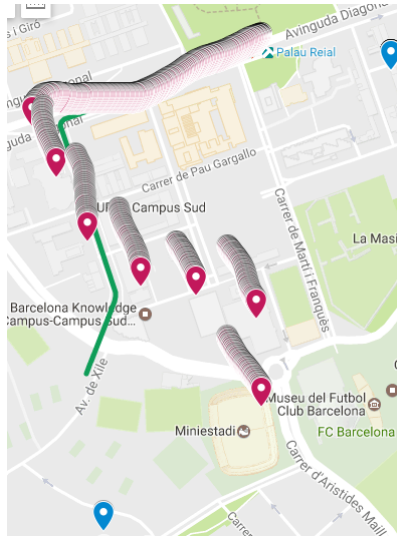


Figura 10.8: Solución Kalman Red Celular/INS (Verde: real; Morado: Kalman, Azul: Torres celulares)

Al empezar con una localización GPS las observaciones de la red celular se incorporan incrementalmente, por eso, una posible mejora sería reinicializar el filtro de Kalman con una muestra GPS cada un intervalo muy grande, obteniendo el mismo efecto.

En cuanto al error medio de navegación registrado (gráfica 10.9), el filtro siempre es mejor que la solución dada por la red celular. Esto se debe a que el sistema de navegación inercial, pese introducir error, introduce menos error que las torres celulares. El error al utilizar la fusión se reduce en un poco más de un 50 % en la mayoría de casos.

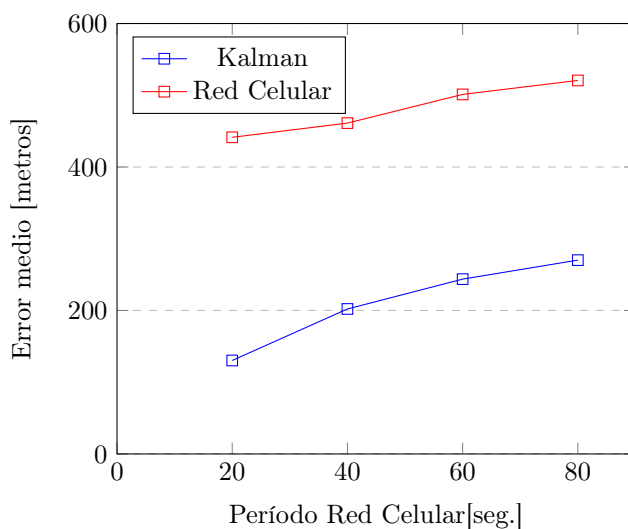


Figura 10.9

10.3. Consumo

El consumo de una aplicación se ha vuelto crítico en los recientes años. Por eso, en esta sección se hace una comparación del consumo de los diferentes métodos estudiados. En la sección 4.4 ya se han visto datos de referencia de consumo, pero en este caso se ha comprobado el consumo de los diferentes tipos de localización utilizando el *BatteryManager*² del SDK de Android y un Samsung Galaxy Mini S3 para hacer las pruebas.

La gráfica 10.10 muestra como el uso del filtro de Kalman introduce muy poco consumo adicional. Así, el uso de la combinación de los sensores inerciales con el GPS aumenta muy poco el consumo respecto al uso solo del GPS. A medida que el intervalo de muestreo aumenta la diferencia de consumo entre el GPS y la fusión se vuelve mínima. El hecho que la diferencia se vuelva mínima no se termina de entender, ya que el uso de los sensores inerciales siempre debería introducir el mismo consumo, pero vemos como también desciende.

²BatteryManager: Clase que permite registrar un receptor para recibir los cambios en el nivel de batería en Android.

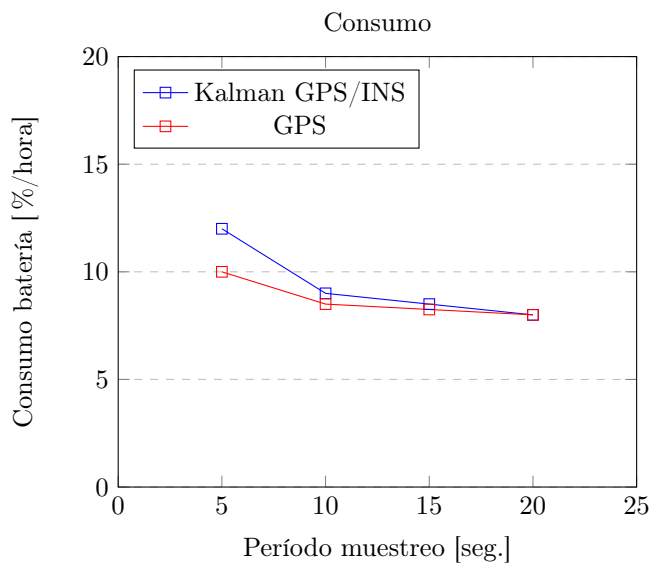


Figura 10.10

También se hicieron pruebas de consumo con la fusión Red Celular/INS. En ese caso se obtuvo un 2 % de consumo para la frecuencia más baja de muestreo, con las demás frecuencias de muestreo el consumo ya está alrededor del 1 %. El aumento de consumo por la integración es mínimo (aproximadamente del 1 %).

11 | Conclusiones

Durante el proyecto se ha estudiado el diseño de un sistema de localización para teléfonos inteligentes que fusione las medidas de los sensores del sistema de navegación inercial(INS) con sensores externos como el GPS o la localización mediante red celular. La aplicación que ha servido de ejemplo es la de un sistema de navegación para un automóvil.

En todo momento se ha intentado mantener los métodos utilizados con la mayor simplicidad y eficacia posible, para así poder reducir o mantener el consumo de potencia de los dispositivos, que es un aspecto de fundamental importancia. .

En cuanto a la integración **GPS/INS**, se ha visto que en el caso del periodo de muestro más bajo (5 seg.), el error medio en la navegación ha resultado ser parecido a el del GPS. Sin embargo, a medida que se ha incrementado el período de muestreo, el filtro ha dado un resultado con un **error más bajo** que el caso de usar únicamente el GPS. Aparte de la precisión del filtro, también se han observado otras propiedad ausentes en la localización con GPS tales como:

- **Robustez** a la introducción de medidas con mucho error, en las simulaciones realizadas, introduciendo valores GPS alejados de la realidad, el filtro no corrige la posición a la recibida si no a un lugar intermedio (introduciendo menos error).
- **Perdida de señal**, en el caso de perder la señal de GPS durante un pequeño intervalo de tiempo, el filtro es capaz de mantener una estimación de la posición (con más error) mediante los sensores inerciales.

El caso más interesante de estudio ha sido la integración **Red Celular/INS**, ya que se ha obtenido una gran mejora respecto al simple uso de la red celular. En este proyecto se ha visto como mediante la integración de éstas y los sensores inerciales se puede mejorar mucho la precisión. Consiguiendo **reducir el error** a la mitad y en algunos trayectos realizados de 400 a 50 metros. Que nosotros sepamos, no hay estudios similares que reporten estas

mejoras significativas. Estos resultados se han obtenido en un entorno urbano, con células de radio reducido, sin duda, en otro tipo de entornos los errores introducidos por la red celular serían mayores (y seguramente los beneficios del método propuesto serían aún mayores).

Para finalizar, el estudio de **consumo energético** ha permitido valorar la viabilidad de las integraciones.

En el caso de el filtro de Kalman GPS/INS, la integración solamente ha supuesto un incremento en el consumo de batería entre el 2 y el 1 %/hora. De este modo, la integración resulta muy viable dado su bajo coste energético.

En el caso del filtro de Kalman Red Celular/INS el consumo sigue siendo mínimo, alrededor del 2 %/hora en el mayor de los casos.

En definitiva, en cualquiera de los dos casos (GPS o redes celulares) el uso de este filtro de fusión de sensores resulta en un incremento mínimo en el consumo energético, debido también al uso de la frecuencia más pequeña de muestreo de los sensores inerciales.

A modo de resumen en la siguiente gráfica se muestran los dos mejores métodos según su consumo:

- Para los casos en los que es imprescindible un bajo consumo energético, la fusión Red Celular/INS permite un error razonable de 130 metros con un consumo de batería en torno al 2 % .
- Por otro lado, cuando usamos el sistema /GPS , podemos tener errores en torno a 50 metros con un consumo del 8 % por hora. Si queremos llegar a un error de 30 metros el consumo alcanza el 12 % .

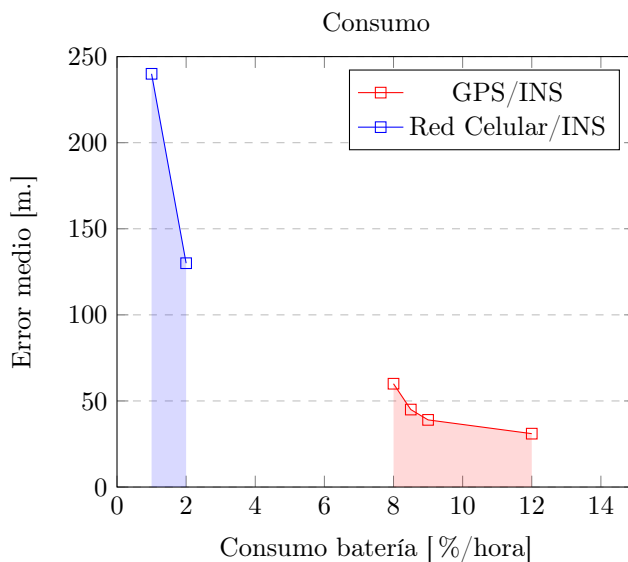


Figura 11.1

En conclusión, en este proyecto se han demostrado los beneficios en cuanto a precisión en la localización y consumo de potencia del uso de la fusión de sensores inerciales con localización por GPS o por redes celulares. Además, tanto el INS como el filtro de Kalman son fácilmente aplicables en cualquier dispositivo Android utilizando el diseño realizado en el proyecto.

12 | Trabajo futuro

Entre las posibles ampliaciones del proyecto encontramos la posibilidad de utilizar otro método de fusión de sensores. Una posibilidad podría ser utilizar el método de Madgwick [12], comúnmente utilizado en el mundo de la navegación de drones. Éste, utiliza los propios sensores de la IMU y un magnetómetro para realizar una fusión eficiente. Existe la posibilidad de poder utilizarlo ya que se obtienen buenos resultados con frecuencias bajas de muestreo. Probablemente, el resultado de este método dependería bastante más de la calidad de los sensores de los dispositivos móviles que el filtro de Kalman, que utiliza medidas externas (GPS o torres celulares).

También está en mente utilizar un filtro de Kalman para unir el proyecto de movilidad con el de calidad del aire del grupo de investigación. Para así poder implementar métodos de fusión de sensores para derivar valores de contaminación según la posición de la gente o derivar otra información.

13 | Valoración Personal

La realización de un proyecto de final de grado siempre merece respeto dada su importancia. Por eso, al hacer este proyecto me he encontrado con obstáculos, pero con esfuerzo y perseverancia he podido superarlos.

El hecho de enfrentarse a la realización de un proyecto de importancia y de más larga duración de lo que estamos acostumbrados, es una experiencia vital para la formación de un ingeniero. De este modo, al enfrentarme a las diferentes etapas que comprende un proyecto (desde planificación a la redacción de la memoria) he mejorado diferentes cualidades, como el aprendizaje autónomo (al tener que aprender conceptos de forma auto-didáctica), el rigor en la experimentación y en la definición de conceptos, la correcta redacción de un documento técnico, etc.

Creo que la realización de este proyecto me ayudará para afrontar con más experiencia proyectos futuros, poniendo en prueba así las cualidades aprendidas. Pese a ser una etapa que requiere de mucho esfuerzo, es aconsejable que todo futuro ingeniero se encuentre con la realización de un proyecto así.

Glosario

Acelerómetro Sensor que mide las fuerzas ejercidas sobre un dispositivo.

Azimut Ángulo que forma la dirección del dispositivo con el norte magnético.

Filtro de Kalman Método clásico de estimación Bayesiana para la fusión de sensores. Técnica utilizada para realizar las fusiones vistas.

Giroscopio Sensor que mide la velocidad angular del movimiento de un dispositivo.

GPS Método muy utilizado que usa la información recibida por un conjunto de satélites para localizar al usuario.

GPS/INS Fusión de las medidas de un GPS con el sistema de navegación inercial (INS).

INS *Inertial Navigation System* o sistema de navegación inercial, consiste en la estimación de posición y dirección utilizando sensores inerciales (acelerómetro y giroscopio).

Magnetómetro Sensor que mide el campo magnético, permitiendo calcular la dirección respecto al norte.

NED Sistema de coordenadas de navegación con ejes en dirección Norte, Este y centro de la Tierra.

Red celular Método de localización que utiliza la red de telefonía móvil.

Red Celular/INS Fusión de la localización mediante la red celular y el sistema de navegación inercial.

Sistema geodésico o LLA Sistema de coordenadas utilizado por el GPS, con las componentes longitud, latitud y altitud.

Anexo : Gantt

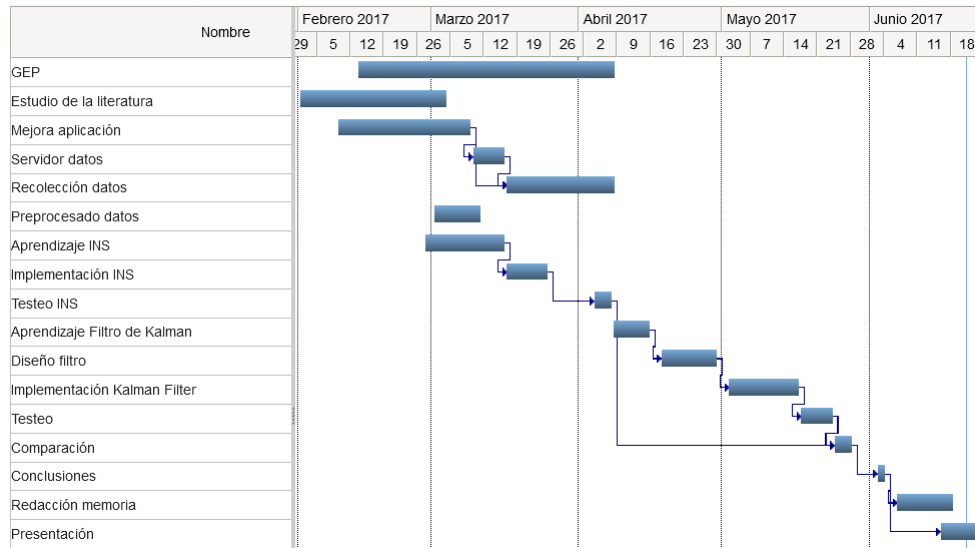


Figura 1Diagrama Gantt

A | Elipses de Error

Un método gráfico para expresar la incertidumbre estadística son las elipses de error. En el caso de la navegación nos permiten visualizar la incertidumbre en la posición dada la media y covarianza de la posición.

La elipse de error codifica la máxima y mínima desviación y sus direcciones. La dirección de máxima varianza se calcula de la siguiente manera:

$$\theta = 0,5 \cdot \tan^{-1} \left[\frac{1}{\text{aspectratio}} \cdot \frac{2\sigma_{xy}}{(\sigma_x)^2 - (\sigma_y)^2} \right]$$

Para obtener una elipse con el 95 % de confianza se utiliza un factor de escala de 2.4477. Finalmente la longitud de los ejes de máxima y mínima varianza vienen dados por:

$$\begin{aligned} L_{max} &= \sqrt{\max(\lambda_0, \lambda_1)} \cdot 2,4477 \cdot 2 \\ L_{min} &= \sqrt{\min(\lambda_0, \lambda_1)} \cdot 2,4477 \cdot 2 \end{aligned}$$

Donde λ_0, λ_1 son los valores propios de la matriz de covarianza.

B | Transformación de coordenadas inversa

Durante la memoria se han visto diferentes transformaciones de coordenadas (body a NED y geodésico a NED) esenciales para poder integrar las medidas de los sensores. En este capítulo del apéndice se muestran las transformaciones inversas, de NED a body y de NED a geodésico.

NED a geodésico: esta transformación es importante para poder mostrar el estado del filtro en un mapa en coordenadas geodésicas (*Google Maps*). Primero se debe realizar una transformación de NED a ECEF.

$$\begin{bmatrix} X_{ECEF} \\ Y_{ECEF} \\ Z_{ECEF} \end{bmatrix} = \begin{bmatrix} -\sin(lat) \cdot \cos(lon) & -\sin(lat) \cdot \sin(lon) & \cos(lat) \\ -\sin(lon) & \cos(lon) & 0 \\ -\cos(lat) \cdot \cos(lon) & -\cos(lat) \cdot \sin(lon) & -\sin(lat) \end{bmatrix}^T \cdot \begin{bmatrix} N \\ E \\ D \end{bmatrix} + \begin{bmatrix} X_{ECEF}^0 \\ Y_{ECEF}^0 \\ Z_{ECEF}^0 \end{bmatrix}$$

Luego, se debe realizar una transformación de ECEF a geodésico, pero esta transformación no tiene formula cerrada, pudiéndose aplicar el siguiente

algoritmo entre otros:

```

Result: lat,lon,alt
lon = atan(y,x);
r =  $\sqrt{x^2 + y^2 + z^2}$ ;
p =  $\sqrt{x^2 + y^2}$ ;
while Cambio en alt do
    | N =  $\frac{R}{\sqrt{1-(N_e^2 \cdot \sin(lat)^2)}}$ ;
    | alt =  $\frac{p}{\cos(lat)} - N$ ;
    | lat =  $atan(\frac{z}{p} \cdot \frac{1}{(1-(N_e^2 \cdot \frac{N}{N+alt})})}$ ;
end
    
```

Algorithm 1: Transformación ECEF a Geodésico

NED a body frame: la transformación de NED a body frame se realiza utilizando la matriz para el ángulo de Euler yaw:

$$\begin{bmatrix} X_b \\ Y_b \\ Z_b \end{bmatrix} = \begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} N \\ E \\ D \end{bmatrix}$$

Bibliografía

- [1] Md Syedul Amin, Mamun Bin Ibne Reaz, Mohammad Arif Sobhan Bhuiyan, and Salwa Sheikh Nasir. Kalman filtered gps accelerometer-based accident detection and location system: A low-cost approach. *Curr. Sci*, 106:1548–1554, 2014.
- [2] Guowei Cai, Ben M Chen, and Tong Heng Lee. *Unmanned rotorcraft systems*. Springer Science & Business Media, 2011.
- [3] Cindy Cappelle, Denis Pomorski, and Y Yang. Gps/ins data fusion for land vehicle localization. In *Computational Engineering in Systems Applications, IMACS Multiconference on*, volume 1, pages 21–27. IEEE, 2006.
- [4] Pavel Davidson, Jani Hautamäki, and Jussi Collin. Using low-cost mems 3d accelerometer and one gyro to assist gps based car navigation system. In *Proceedings of 15th Saint Petersburg International Conference on Integrated Navigation Systems (May 2008)*, 2008.
- [5] Wilfried Elmenreich. An introduction to sensor fusion. *Vienna University of Technology, Austria*, 2002.
- [6] Kenneth Gade. Introduction to inertial navigation and kalman filtering. In *Tutorial for IAIN World Congress, Stockholm*, pages 1–57, 2009.
- [7] Denis Gingras. An overview of positioning and data fusion techniques applied to land vehicle navigation systems.
- [8] B. Gurram and N. Giri. Improving localization accuracy of android’s fused location provider api using kalman filter. In *2016 International Conference on Computer Communication and Informatics (ICCCI)*, pages 1–4, Jan 2016.
- [9] Bhaskar Gurram and Nupur Giri. Improving localization accuracy of android’s fused location provider api using kalman filter. In *Computer Communication and Informatics (ICCCI), 2016 International Conference on*, pages 1–4. IEEE, 2016.

- [10] Fredrik Gustafsson, Fredrik Gunnarsson, Niclas Bergman, Urban Forssell, Jonas Jansson, Rickard Karlsson, and P-J Nordlund. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on signal processing*, 50(2):425–437, 2002.
- [11] Zhang Liu. Introduction to inertial navigation and pointing control. *Northrop Grumman Corporation, California, USA*, 2011.
- [12] Sebastian Madgwick. An efficient orientation filter for inertial and inertial/magnetic sensor arrays. *Report x-io and University of Bristol (UK)*, 25, 2010.
- [13] Niklas Magnusson and Tobias Odenman. Improving absolute position estimates of an automotive vehicle using gps in sensor fusion. 2012.
- [14] E Nebot, S Sukkarieh, and H Durrant-Whyte. Inertial navigation aided with gps information. In *Mechatronics and Machine Vision in Practice, 1997. Proceedings., Fourth Annual Conference on*, pages 169–174. IEEE, 1997.
- [15] Rudy Negenborn. *Robot localization and Kalman filters*. PhD thesis, Utrecht University, 2003.
- [16] BA Ragel and M Farooq. Comparison of forward vs. feedback kalman filter for aided inertial navigation system. In *Information Fusion, 2005 8th International Conference on*, volume 1, pages 6–pp. IEEE, 2005.
- [17] Sławomir Romaniuk and Zdzisław Gosiewski. Kalman filter realization for orientation and position estimation on dedicated processor. *acta mechanica et automatica*, 8(2):88–94, 2014.
- [18] Simo Särkkä. Bayesian estimation of time-varying systems: Discrete-time systems. *Written material for the course S-114.4202, Aalto University, School of Science and Technology*, 2011.
- [19] Sebastian Thrun, Wolfram Burgard, and Dieter Fox. *Probabilistic robotics*. MIT press, 2005.
- [20] Greg Welch and Gary Bishop. An introduction to the kalman filter. 1995.
- [21] Oliver J Woodman. An introduction to inertial navigation. Technical report, University of Cambridge, Computer Laboratory, 2007.

- [22] Feihu Zhang, Hauke Stähle, Guang Chen, Chao Chen Carsten Simon, Christian Buckl, and Alois Knoll. A sensor fusion approach for localization with cumulative error elimination. In *Multisensor Fusion and Integration for Intelligent Systems (MFI), 2012 IEEE Conference on*, pages 1–6. IEEE, 2012.